

1993

Effects of structural elements on vertical axis wind turbine power production

Edward Panek

San Jose State University

Follow this and additional works at: https://scholarworks.sjsu.edu/etd_theses

Recommended Citation

Panek, Edward, "Effects of structural elements on vertical axis wind turbine power production" (1993). *Master's Theses*. 639.

DOI: <https://doi.org/10.31979/etd.22vj-v7pa>

https://scholarworks.sjsu.edu/etd_theses/639

This Thesis is brought to you for free and open access by the Master's Theses and Graduate Research at SJSU ScholarWorks. It has been accepted for inclusion in Master's Theses by an authorized administrator of SJSU ScholarWorks. For more information, please contact scholarworks@sjsu.edu.

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.



University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313: 761-4700 800: 521-0600

Order Number 1354154

**Effects of structural elements on vertical axis wind turbine
power production**

Panek, Edward, M.S.

San Jose State University, 1993

U·M·I

300 N. Zeeb Rd.
Ann Arbor, MI 48106

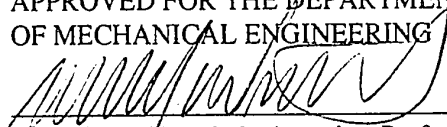
EFFECTS OF STRUCTURAL ELEMENTS ON VERTICAL AXIS WIND TURBINE POWER PRODUCTION

A Thesis Presented to
The Faculty of the Department of Mechanical Engineering
San Jose State University

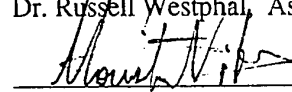
In Partial Fulfillment
of the Requirements for the Degree
Master of Science

By
Edward Panek
August, 1993

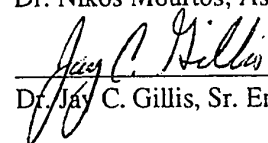
APPROVED FOR THE DEPARTMENT
OF MECHANICAL ENGINEERING



Dr. Russell Westphal, Associate Professor

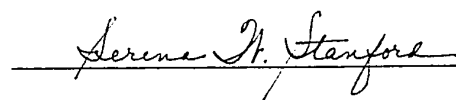


Dr. Nikos Mourtos, Associate Professor



Dr. Jay C. Gillis, Sr. Engineer S. Levy, Inc.

APPROVED FOR THE UNIVERSITY



ABSTRACT

EFFECTS OF STRUCTURAL ELEMENTS ON VERTICAL AXIS WIND TURBINE POWER PRODUCTION

by Edward Panek

In the past decade vertical axis wind turbines, VAWTs, have proven to be viable electric power producing machines at the Altamont Pass in California. Various performance models have been established to analyze these turbines but certain structural retrofits that degrade power output have not yet been thoroughly studied. Interference drag where a blade support strut joins with the turbine blade is one case. Recently developed blade patches designed to strengthen the splices where blade segments come together is a second case. A streamtube model is used to assess the affects of strut joints and blade patches on wind turbine performance. Measured lift and drag data for both the blade patches and the strut joints are incorporated into the analysis to predict power losses relative to a bare rotor. Data for faired blade patches and strut joints is also used to predict the amount of power that can be recovered if fairings are employed.

TABLE OF CONTENTS

I	Introduction	
	Modern Wind Power	1
	Flowind Vertical Axis Wind Turbines	5
	Objective	7
II	Analysis	
	VAWT Performance Models	9
	Multiple Streamtube Model	12
	Streamtube Theory	13
	Blade Patch and Strut Joint Lift & Drag Data	19
III	Test Cases	26
IV	Results	30
V	Conclusions	35
VI	References	37
Appendices		
A	MSVAWT Computer Program Listing	A 1
B	Sandia 5 Meter Test Case Input	B 1
C	Flowind 19 Meter Test Case Input	C 1

LIST OF FIGURES

- 1 Water Pumping Dutch Windmill
- 2 Horizontal Axis Wind Turbine
- 3 Early Persian Windmill
- 4 Savonius Wind Turbine
- 5 Power Coefficient vs. Tip Speed Ratio for Common Wind Machines
- 6 Sandia 17 Meter VAWT
- 7 Flowind 19 Meter Turbine
- 8 Cross Section of Original and Modified Splice for a Flowind VAWT
- 9 Typical Streamtube with Wind shear Distribution and Average Force Acting on the Blade Element
- 10 Normal and Tangential Forces
- 11 Forces on the Blade Element
- 12 Relative Velocity Components
- 13 Cross Section of a Blade Splice for a Flowind VAWT
- 14 Lift and Drag Coefficients for a Blade Patch
- 15 Faired Blade Splice for a Flowind VAWT
- 16 Lift and Drag Coefficients for a Faired Blade Patch
- 17 Drag Coefficient for a Strut Joint
- 18 Drag Coefficient for a Faired Strut
- 19 Geometry for a Sandia 5 Meter Test Case
- 20 Power vs. Wind Speed for a Sandia 5 Meter Test Case
- 21 Bare Rotor Geometry for a Flowind 19 Meter Test Case
- 22 Power vs. Wind Speed for a Flowind 19 Meter Test Case
- 23 Power Curves for a Blade Patch on a Flowind 19 Meter Turbine and a Typical Wind Distribution
- 24 Power Curves for a Strut Joint on a Flowind 19 Meter Turbine
- 25 Fraction of Annual Energy Production for a Flowind 19 Meter Turbine with Blade Patches and Strut Joints

List of Tables

- 1 Drag Data for Blade Splice
- 2 Drag Data for Strut Joint
- 3 Percent of Annual Energy Production for a Flowind 19 Meter Turbine
with Blade Patches and Strut Joints

NOMENCLATURE

\bar{a}	Interference factor
A_s, A_p	Area of a streamtube, plan view area of the blade element
A_T	Turbine swept area
C_l, C_d	Lift, drag coefficients
C_p	Power Coefficient
\bar{f}_s	Average or effective force in the streamtube
f_s	Force in the streamtube
f_n, f_t, f_l, f_d	Normal, tangential, lift, drag forces on the blade element
f^*	Dimensionless streamtube force
h or $h(W)$	Wind frequency distribution
k	Von Karman constant
N_b	Number of blades
N_θ	Number of θ divisions
P or $P(W)$	Power
R or $R(z)$	Radius of turbine rotor
Re	Chord Reynolds number
S	Speed of blade element
T_z	Torque on a vertical of the rotor
T	Total torque on the rotor
U_1, U_2	Upwind streamtube air speed, downwind streamtube air speed
U	Air speed inside rotor
u_{fric}	Friction velocity
v	Relative velocity
v_t, v_n	Relative velocity components
W or $W(z)$	Wind speed
W_{ref}, V_∞	Reference wind speed

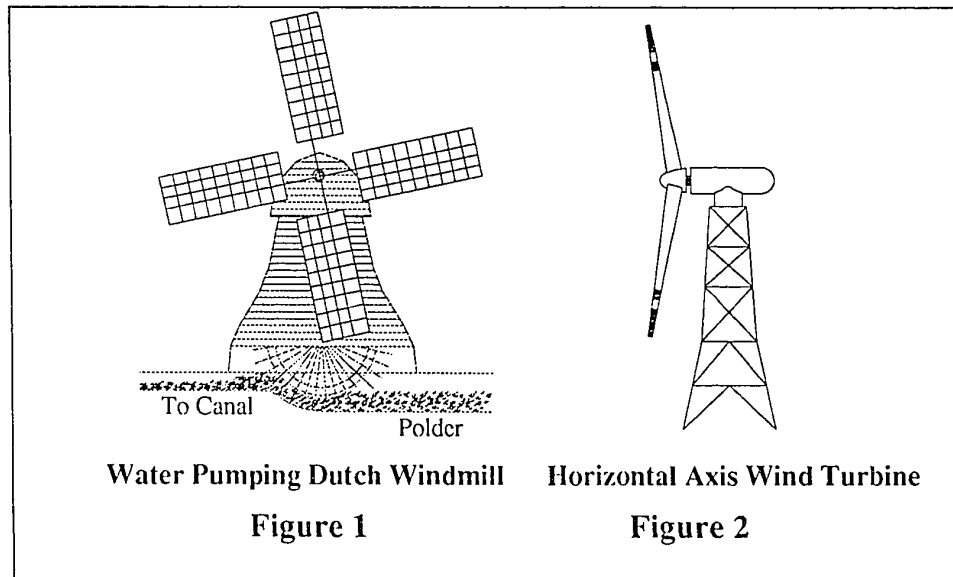
X	Tip Speed Ratio
z	Vertical distance
z_o	Friction height
z_{ref}	Reference height for wind speed

α	Angle of attack
θ	Rotation angle of rotor
ρ	Air density
ω	Rotor rate of rotation

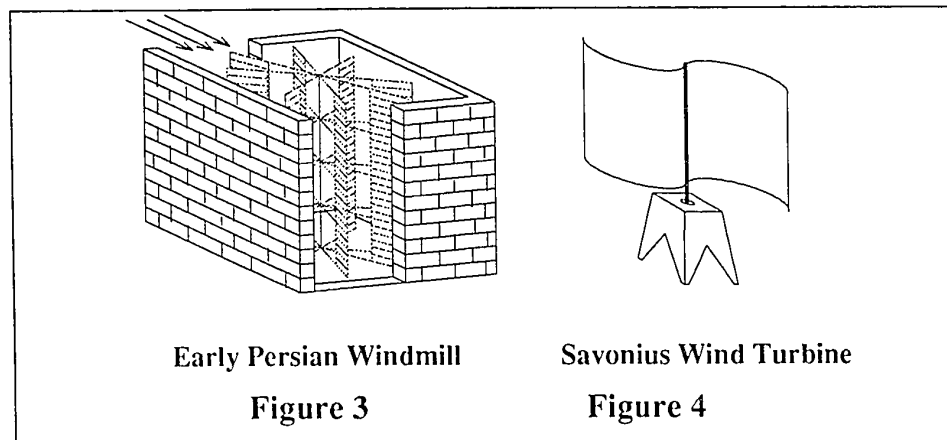
I INTRODUCTION

MODERN WIND POWER STATE OF THE ART DARRIEUS WIND TURBINES

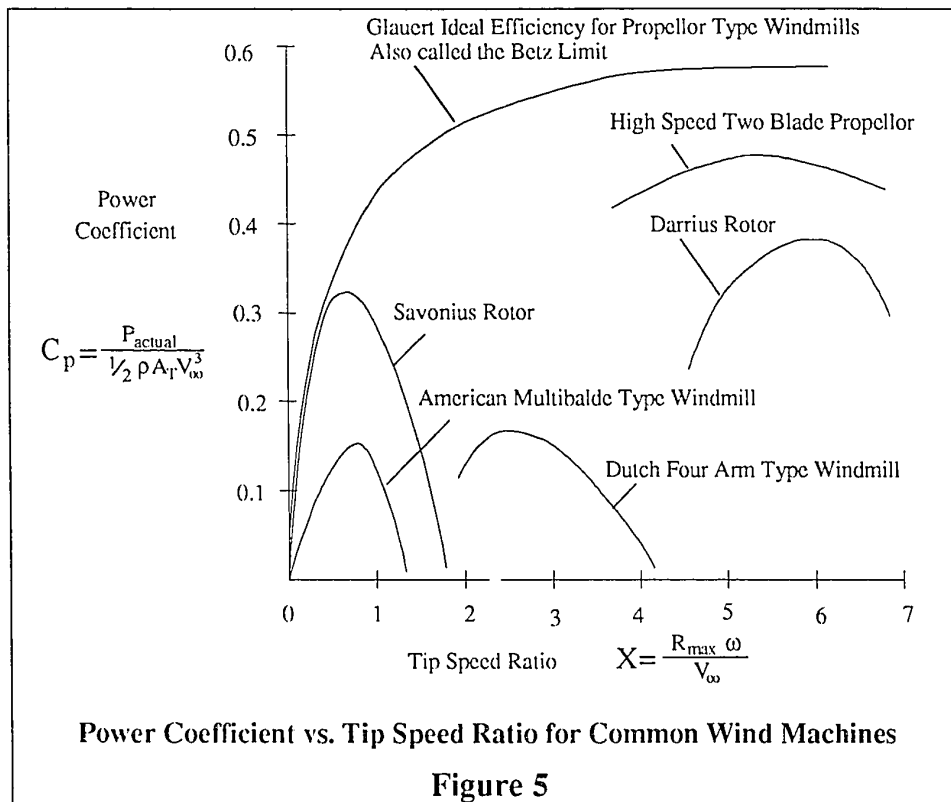
Modern wind power machines take one of two forms: they are either horizontal or vertical axis type. Horizontal axis wind turbines are similar to their windmill predecessors shown in figure 1. The major difference is that modern horizontal axis machines, shown in figure 2, have fewer and much better aerodynamically designed blades. This allows the modern machines to operate at much higher rotation rates for a given blade diameter. The higher rotation rate is an advantage when generating power because one can better match electrical generator characteristics.



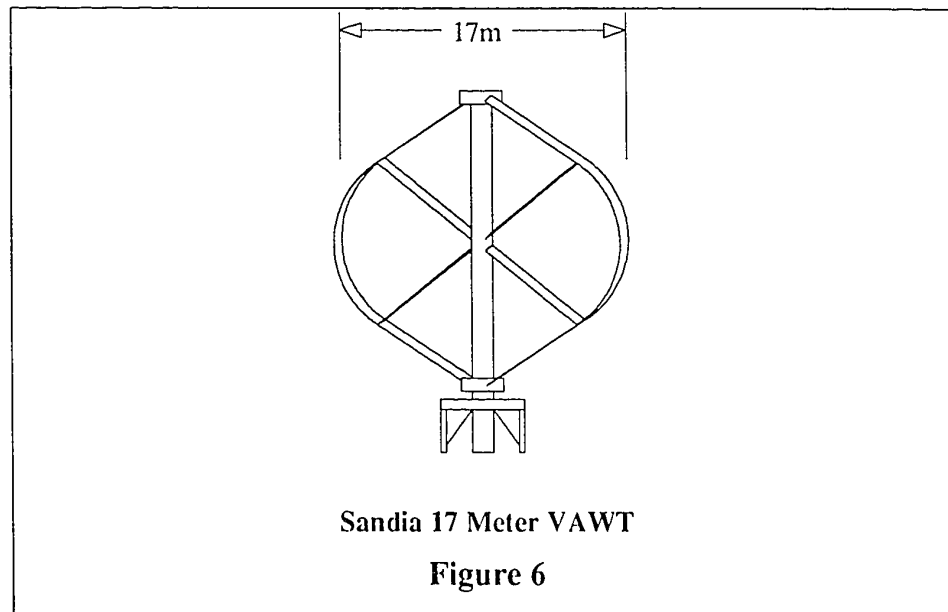
Vertical axis wind turbines, at least in appearance, are more akin to early Persian windmills [1] shown in figure 3. The Persian windmills, however, had limitations because the slots in the tower needed to be aligned with the wind direction. Modern vertical wind turbines do not have this restriction - they operate independent of wind direction.



In 1920 a Finn named S. J. Savonius experimented with a cup shaped sail on a vertical axis. His design is pictured in figure 4. This shape created a machine that could produce power independent of wind direction. At that time airfoils and aircraft propellers were the popular pursuit among engineers. Perhaps due to this influence, a French Monsignor named Georges Jean Darrieus in 1929 substituted two long airfoil shapes in place of the sails on a vertical axis wind machine. This new idea used the lift characteristic of an airfoil rather than the drag characteristic applied in the Savonius or Persian designs [2]. The Darrieus turbine became the inspiration for modern vertical axis wind turbines, or VAWTs, as they are often referred to. Performance curves for various wind machines are shown in figure 5 [3].



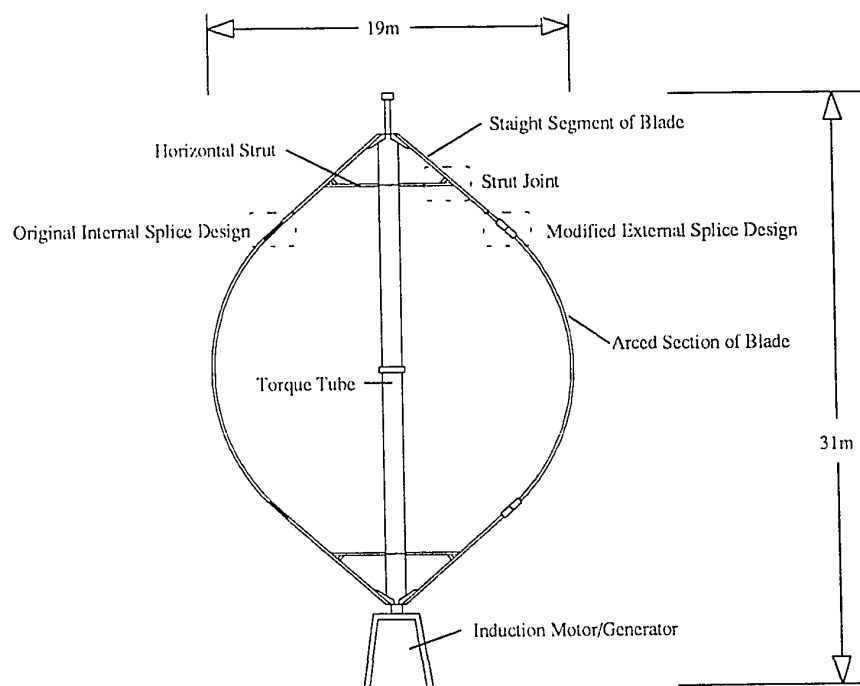
In the early 1970's two engineers, Peter South and Raj Rangi, working for Canada's National Research Council built what is now recognized as a modern Darrius Turbine [4]. Thin metal airfoils in the shape of an arc attached to the ends of a vertical axis. The "eggbeater" shape was not as efficient in producing power for a given swept area as the better studied horizontal axis machines but the design was simpler. Shortly after this, some interested NASA engineers fabricated a Darrius turbine of their own. The machine accidentally caught the eye of some media people who showed footage of the machine on TV. This generated a swell of public interest. Perhaps this interest produced support for wind turbines at the Department of Energy. They funded the Sandia Laboratories to design and build vertical axis wind turbines for study.



The Sandia engineers designed and fabricated three different VAWTs between 1974 and 1977. The three machines had approximate diameters of 2, 5 and 17 meters. When testing had finished the 17 meter design, illustrated in figure 6, had progressed to a viable and practical power producing machine. The 17 meter design consisted of two aluminum blades each comprised of an arced segment connected to two straight segment. Struts, emanating from the middle of the torque tube, attached to the turbine blade where the arced and straight segments of the blade came together. Both the blades and the struts had NACA0012 airfoil cross sections. A modified version of this 17 meter VAWT was later produced by Flowind Corporation in California [5]. A 19 meter design very similar to the original Sandia 17 meter VAWT was also produced by Flowind. VAWTs are operated by Flowind at both the Altamont Pass and at Tehachepi in California.

FLOWIND VERTICAL AXIS WIND TURBINES

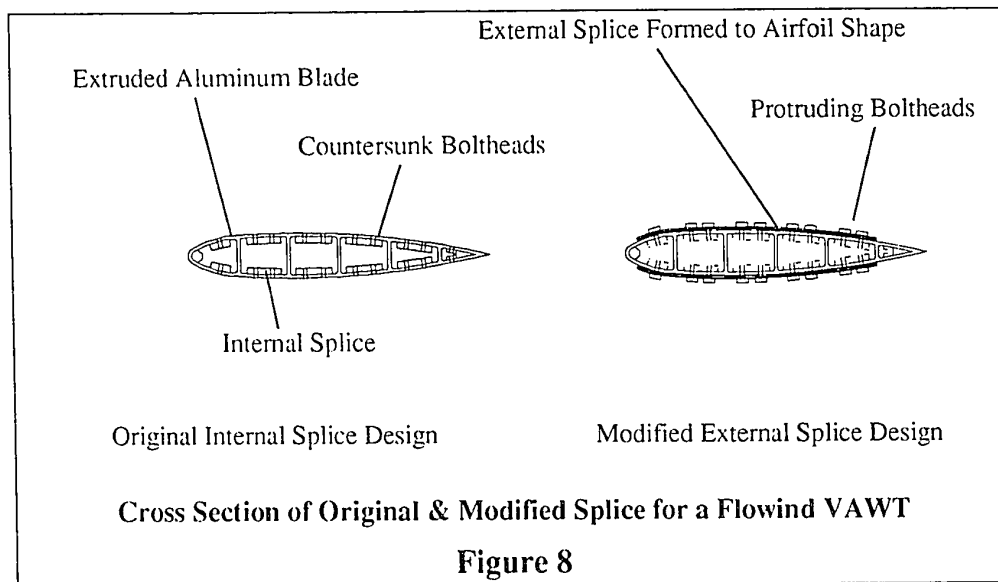
The Flowind 17 and 19 meter VAWTs were slightly different from the original Sandia 17 meter design. The Flowind turbines do not have the struts connected at the arced/straight segments of the blade like the Sandia turbines. Instead, the Flowind designs have horizontal struts near the bottom and top of the torque tube. Where the arced and straight segments of the blades interface only a mechanical connection remained. Both the 19 and 17 meter designs use a NACA0015 cross section. A Flowind F-19 turbine is shown in figure 7.



Flowind 19 Meter Turbine

Figure 7

After some 10 years of operation the cyclic loads on the aluminum blades began to cause cracks at the mechanical splices between the arced/straight segments of the blades. The original design consists of internal splices within the hollow, extruded blades fastened together with bolts that pass through both the blade and the internal splice. The bolts are countersunk so that they are flush with the surface of the blade. Because of the cracks that occurred in the original design, external splices were developed to reinforce the original splices. The external splices, also referred to as blade patches, consist of two plates formed to the contour of the airfoil shape. The plates are bolted to the surface of the blade segments creating a rough surface. Cross sections of the two splice designs are shown figure 8.



Unfortunately these blade patches compromised some aerodynamic performance to ensure structural integrity. Some rough estimates of the performance loss caused by these patches were available but no computer model able to corroborate these estimates was readily amenable to this problem. A small drop in turbine performance produces a large

percentage decrease in operating profit. A good understanding of the power loss these patches incur is needed so that cost/benefit analysis can be performed when modifying turbines with failed splices.

Interference drag occurs where the horizontal strut connects to the turbine blade. Although it has always been present, the interference drag has not been thoroughly analyzed for Flowind turbines. Assessing the performance effects caused by the strut joints will provide a basic understanding of the power losses associated with the interference drag. A comparison can then be made between the existing joint design and possible modifications, such as fairings, aimed at improving power output.

OBJECTIVE

The objective of this work is to assess the effects of two structural elements on the performance of vertical axis wind turbines: (i) external splices and (ii) blade strut joints. A computer model is developed to predict the output power for a given VAWT geometry as a function of wind speed. The approach is to first predict the power curve for a prospective VAWT geometry without including losses due to blade patches or strut joints. This establishes a nominal power curve akin to what other VAWT performance codes predict. The next step is to determine the power losses caused by the structural components. An assessment of what effect each kind of drag has on power output and annual energy production can then be made.

Because VAWT performance codes do not yet accurately predict the output power for all VAWT geometries over all possible wind speeds a cautious approach is taken when drawing conclusions from the results. What is desired here is not the net power loss but

the percent loss relative to the nominal power curve. With this information one can project the predicted losses from the computer analysis onto known output power curves for wind turbines with some level of assurance. This work asserts that while net power predictions from the computer code may not be as accurate as desired, the percent loss predicted by the computer code will closely approximate the fraction of real power losses incurred by the presence of blade patches and blade strut joints.

II ANALYSIS

VAWT PERFORMANCE MODELS

Several methods for analyzing vertical axis wind turbines exist. Basically there are three kinds of models: streamtube models, vortex models, and models based on the Navier Stokes equations.

Streamtube models, originally applied to VAWTs by R. J. Templin [6], come from Glauert's actuator disk theory [7]. Templin considered the entire turbine to be contained within a single streamtube. Using Glauert's formulation relating averaged forces on the rotor, via lift and drag coefficients, to the momentum change across the turbine Templin produced reasonably accurate power predictions for vertical axis wind turbines. Later J. H. Strickland [8] extended the single streamtube model to a multiple streamtube model so that more realistic estimates of the blade forces could be made and the wind shear effects close to the ground could be included. I. Paraschivoiu [9] applied multiple streamtubes to both the upwind and downwind sides of the rotor hoping to get even better estimates of the blade forces and performance. Paraschivoiu later incorporated a dynamic stall model into his streamtube method [10]. Paraschivoiu also assessed the effects of streamtube expansion [11] and secondary effects such as the rotating torque tube and struts [12].

Because streamtube models appeared limited in their ability to predict the cyclic loads acting on the blades, J. H. Strickland [13] produced a vortex model. This allowed a better assessment of the blade forces. In this method lift and drag coefficients are used to calculate induced velocities over a segment of the rotor blade. Unlike other vortex models, Strickland produced a three-dimensional model capable of analyzing the

complications of a curved rotor blade. He was successful in getting improved estimates of blade forces but overall power predictions were roughly the same as streamtube methods. Strickland's vortex method also included a dynamic stall model.

The steady Navier Stokes equations have also been applied to VAWT analysis [14]. Here again the lift and drag coefficients are used to find forces on the blades. The time averaged forces are modeled as momentum sources and sinks in the NS equations. The ability to predict turbine performance was comparable to streamtube and vortex methods.

Note that in all the models described above the section lift and drag data are essential. The results from any of these methods are only as good as this input data. This can partially explain why the three methods each predict about the same power output. When one sums and averages all the forces over the entire turbine blade as it completes one rotation the differences tend to wash out and the predicted output power is essentially the same. So if one wants only to predict power output any method will suffice. The accuracy of the result will be linked to the quality of the lift and drag data. Predictions of the time-dependent forces on the rotor blades, on the other hand, are influenced by the method used. This is because different models have different inherent assumptions. Streamtube methods assume a quasi-steady flow, a constant streamwise velocity in the vicinity of the rotor and that there is no velocity component normal to the free stream flow. Strickland's three-dimensional vortex model does not make all of these assumptions and therefore gives better estimates of the local flow and time-dependent forces on a blade element. Still, vortex methods are only as good as the input lift and drag data.

The lift and drag coefficients used in any of these methods come from experiments done in a steady flow using two-dimensional models. The blades of a vertical axis wind turbine do not, however, operate in a steady flow. Each blade "sees" a cyclically changing flow as it

advances (moves into the wind) and then retreats (moves with the wind). The "true" unsteady lift and drag coefficients have not been determined for such dynamic conditions. A common assumption made, for lack of anything better, is that for small angles-of-attack the rotor blades encounter forces that are essentially equal to those measured under steady state conditions. But for VAWTs, the angle-of-attack changes significantly in time as the rotor sweeps through an entire revolution in the wind flow field. Just how much this dynamic condition affects the "true" instantaneous lift and drag coefficients is not thoroughly understood.

Gormont developed an empirical dynamic stall model used in helicopter analysis [15], but this model was developed for small changes in angle-of-attack. Paraschivoiu based his dynamic stall model on the Gormont theory even though VAWTs experience much greater changes in angle-of-attack. Note that the Gormont theory was developed for thin airfoils; VAWT rotors use thick airfoils in their design. Note also that dynamic stall is usually studied in a two-dimensional environment. The effects of the three-dimensional VAWT blades are unknown. Still, Paraschivoiu found improvement in the performance prediction when a dynamic stall model was incorporated.

A thorough understanding of dynamic stall effects is not yet complete although new apparatus has recently been developed for such studies [16]. Work at Ohio State University, under contract by Sandia National Laboratories, may also help answer the dynamic stall question[17].

MULTIPLE STREAMTUBE MODEL

The first part of this analysis is the streamtube theory. A *double multiple* streamtube method was chosen for this analysis because of its simplicity and relatively small computer time demands compared to other methods. The streamtube theory pursued in this work follows from the multiple streamtube work of Strickland where the projected area of the turbine is broken down into "multiple" horizontal and vertical slices. The model developed here differs from Strickland's work in that it breaks the streamtube into an upwind and a downwind section. This results in the "double" multiple streamtube analysis akin to Paraschivoiu.

Interference drag and power losses due to external splices are accounted for by using appropriate lift and drag coefficients determined by Westphal et al [18][19]. This lift and drag data is not complete for the entire range of Reynolds numbers and angles-of-attack required to analyze the standard operating conditions of a Flowind VAWT. Extrapolating the available lift and drag data for the structural elements is the second part of the analysis. This ad hoc extrapolation is really more art than analysis. Very simple assumptions were made in developing the extrapolated lift and drag data.

As noted earlier the multiple streamtube theory is only as good as the lift and drag data used in the computations. The lack of complete data should cause some skepticism if the results are interpreted without some reference to the known operating conditions of vertical axis turbines. The lift and drag data developed to represent the patched and joint sections is meant to assess the relative significance of these structural elements.

STREAMTUBE THEORY

The so called double multiple streamtube theory used here is based on a simple streamwise momentum model that equates the streamwise forces on the blades to the change in momentum through the streamtube. This is done both on the upwind side and downwind side of the rotor. From Glauert's simple momentum theory one finds that for a single streamtube

$$\bar{f}_s = 2 A_s \rho (W - U1) U1 \quad (1)$$

The density ρ remains nearly constant. $U1$ is the upwind rotor streamtube velocity derived from airscrew theory. It is the effective velocity acting on the blade element and is equal to the average of the upstream and downstream windspeeds, $U1 = (W + U)/2$. A_s is the streamtube area shown in figure 9.

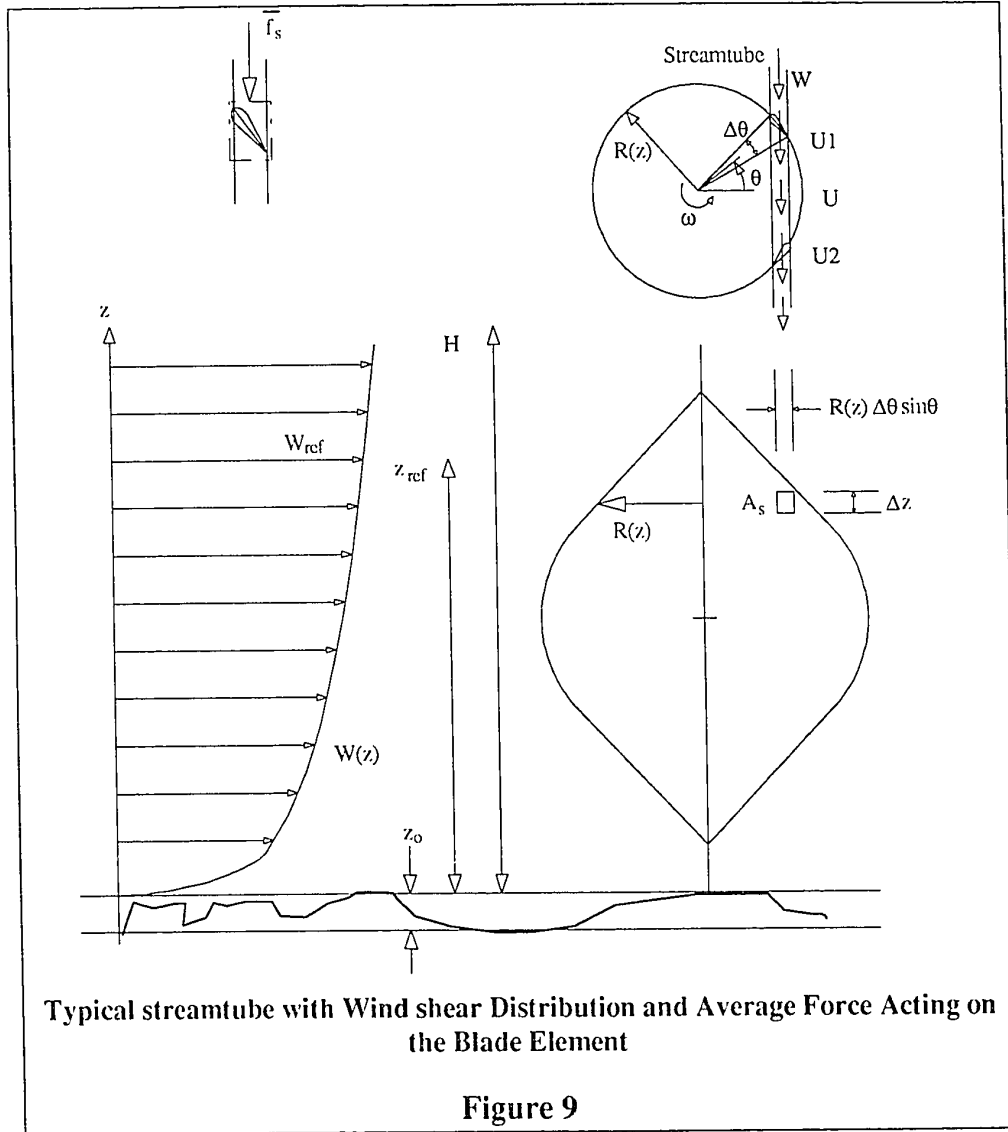
$$A_s = R(z) \Delta\theta \sin \theta \Delta z \quad (2)$$

W is the wind speed at height z [20]:

$$W(z) = (u_{fric} / k) \ln \left(\frac{z + z_o}{z_o} \right) \quad (i)$$

$$u_{fric} = \frac{k W_{ref}}{\ln(z_{ref} / z_o)} \quad (ii)$$

where k is Von Karman's constant, z_o is friction height, and z is the distance above the friction height. Here W_{ref} is the reference wind speed at reference height z_{ref} . For the work here z_{ref} equals 10m as suggested by Brown. Equations i & ii represent the logarithmic wind shear distribution derived from the law of the wall where the shear stress is assumed constant near the boundary.

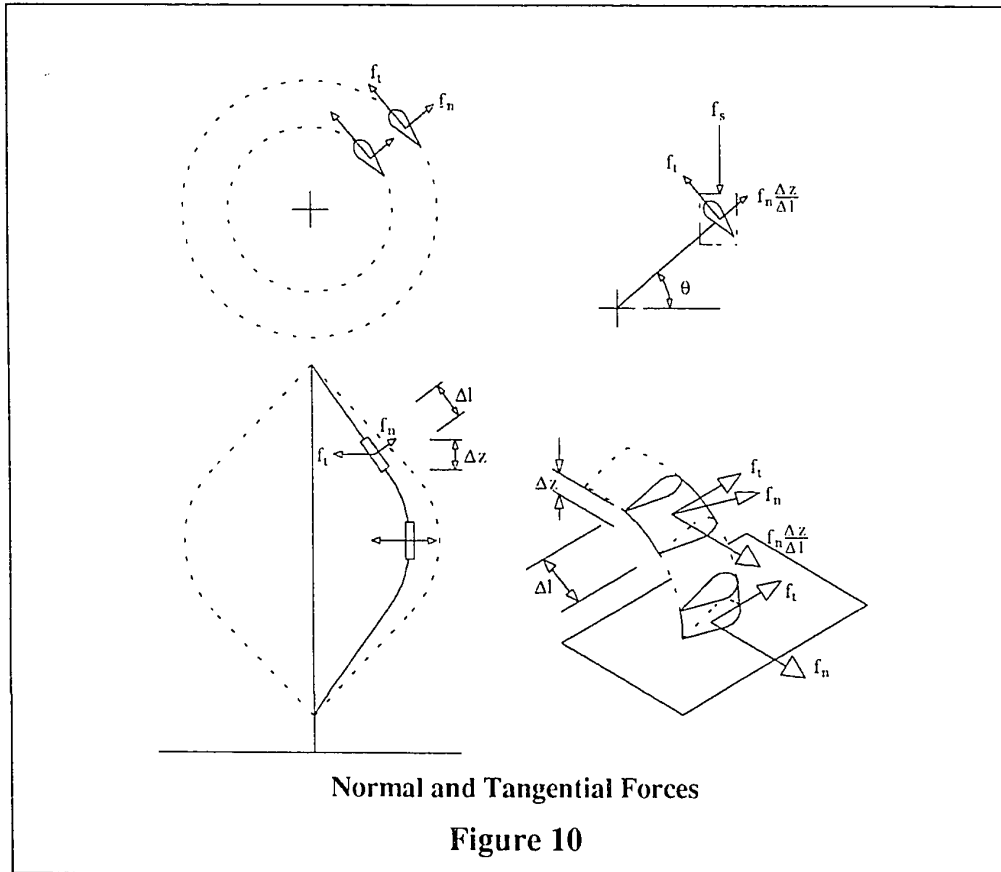


The average force in the streamtube is \bar{f}_s . This time-averaged force is a fictitious, quasi-steady force that represents the momentum change that occurs as the air in the streamtube passes over the blade element. Noting that each blade element spends $\frac{\Delta\theta}{2\pi}$ of its time in each streamtube and that there are N_b number of blades one obtains

$$\overline{f_s} = N_b \frac{\Delta\theta}{2\pi} f_s \quad (3)$$

where f_s represents the streamwise force necessary to maintain a blade element in a state of static equilibrium. Equations (1) & (3) are the cornerstone of streamtube theory. Together they form the nexus that equates the quasi-steady momentum change in the streamtube to the forces on the blade element. The streamwise force f_s is derived from the normal and tangential force components as shown in figure 10.

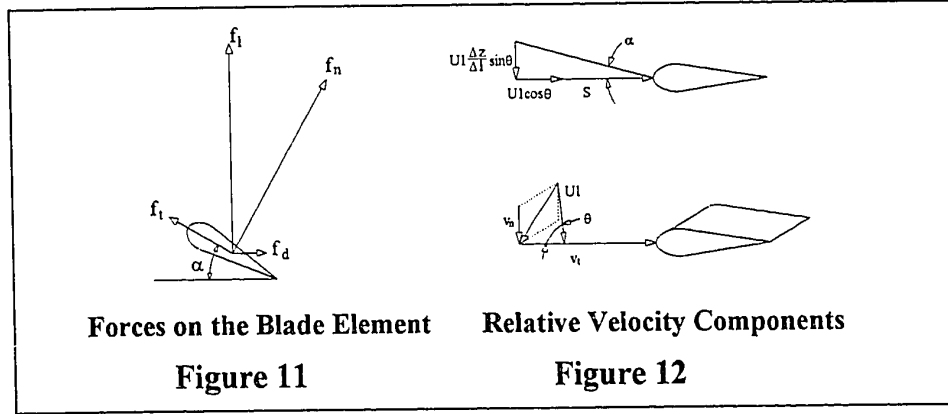
$$f_s = f_n \left(\Delta z / \Delta l \right) \sin \theta + f_t \cos \theta \quad (4)$$



The normal and tangential components are derived from the lift and drag forces on the blade element. These are shown in figure 11.

$$f_n = f_l \cos \alpha + f_d \sin \alpha \quad (5)$$

$$f_t = f_l \sin \alpha - f_d \cos \alpha \quad (6)$$



The lift and drag forces are calculated from the two-dimensional section lift and drag coefficients C_l & C_d :

$$f_l = C_l \frac{1}{2} \rho v^2 A_p \quad (7)$$

$$f_d = C_d \frac{1}{2} \rho v^2 A_p \quad (8)$$

A_p is the plan view area of airfoil section. The airspeed v and the angle-of-attack α relative to the blade element are found from the rotor streamtube velocity $U1$, the tip speed $S = R(z)\omega$ and the rotation angle θ ; see figure 12.

$$v_t = U1 \cos \theta + S \quad (9)$$

$$v_n = U1 (\Delta z / \Delta l) \sin \theta \quad (10)$$

$$v = \sqrt{v_t^2 + v_n^2} \quad (11)$$

$$\alpha = \arctan(v_n / v_t) \quad (12)$$

By rearranging (1) and dividing by W^2 we have the dimensionless streamtube force:

$$f^* = \frac{\bar{f}_s}{2A_s \rho W^2} = \left(1 - \frac{U1}{W}\right) \frac{U1}{W} \quad (13)$$

Now define the interference factor according to Strickland:

$$\bar{a} = 1 - \frac{U1}{W} \quad (14)$$

Combining equations (13) and (14)

$$f^* = \bar{a}(1 - \bar{a}) \quad (15)$$

or

$$\bar{a} = f^* + \bar{a}^2 \quad (16)$$

Equation (16) forms a basis to an iterative solution for the streamtube momentum equation. Starting on the upwind side of the rotor one does the following as outlined by Strickland:

- \bar{a} is set equal to zero which gives an initial value for $U1$.
- v and α are obtained from equations (9), (10), (11) and (12).
- With v and α one obtains C_l and C_d from airfoil section data. With these one finds f_t & f_n using equation (7) & (8). This gives f_n & f_t from equations (5) & (6). Now f_s is known from equation (4).
- \bar{f}_s is then calculated with equation (3) and f^* is obtained from equation (13).

- Using the present value of \bar{a} and f^* from (14) & (15) on right hand side of equation (16) one obtains a new value for \bar{a} .
- Now compare the new value of \bar{a} to the old one. If convergence has not been attained repeat the process starting with the new value for \bar{a} . Convergence is achieved when

$$\left| \frac{\bar{a}_{\text{new}} - \bar{a}_{\text{old}}}{1 - \bar{a}_{\text{old}}} \right| < 0.01 \quad \text{or} \quad \left| \frac{U_{1\text{old}} - U_{1\text{new}}}{U_{1\text{old}}} \right| < 0.01 \quad (17)$$

After satisfactory convergence U_1 is known. The tangential force, f_t , which creates the driving torque on turbine, is also known.

The same procedure is followed for the downwind side of the rotor, the only difference is the input free stream velocity $U = 2U_1 - W$ is now used in place of W . The value for U comes from the Glauert theory. The effective turbine streamtube velocity is now U_2 and one follows the same procedure that was outlined for the upwind side of the rotor only now

$$\bar{f}_s \Big|_{\text{downwind}} = 2 A_s \rho (U - U_2) U_2 \quad (18)$$

Once all upwind and downwind forces are known in each streamtube the rotor torque can be calculated. The torque for a slice Δz is the averaged sum of all the tangential force components as θ goes from 0 to 2π

$$T_z = R(z) \frac{N_b}{N_\theta} \sum_{\theta} f_t \quad (19)$$

where N_b is the number of blades and N_θ is the number of θ divisions.

The total torque on the turbine is the sum of all the z-axis slices.

$$T = \sum_z T_z \quad (20)$$

The power is easily found from

$$P = T\omega \quad (21)$$

Using the procedure described above one can determine a power vs. wind speed curve for a given VAWT geometry. The total energy produced during a year can then be calculated if a wind speed frequency distribution is known.

$$\text{Energy Production} = \sum_W P(W)h(W) \quad (22)$$

where $P(W)$ is the power output at a given wind speed and $h(W)$ is the numbers of hours per year a turbine operates at that wind speed.

BLADE PATCH AND STRUT JOINT LIFT AND DRAG DATA

The lift and drag data developed here is derived from the Eppler data for a NACA0015 airfoil [21]. Limited measured drag and stall data for the splice patch and strut joint is available; these measured values are used to modify the Eppler lift and drag curves. The basic approach is to limit the lift data according to simple assumptions and alter the drag data so that it conforms to measured values.

For both the faired and unfaired patch sections one measured value of the drag coefficient at the corresponding stall angle is known for a chord Reynolds number of 1 million. The lift coefficients were not measured for either case, but stall angles were determined. Table

1 summarizes the experimental and Eppler data for an unfaired blade patch. Similarly, the measured drag for the faired and unfaired strut joint is known at just one angle-of-attack for a chord Reynolds number of 1 million. The results are summarized in table 2.

Table 1

Drag Data for a Blade Splice

Re = 1 million	Measured Data for External Splice	Measured Data for Faired Splice	Original Eppler Data
Drag Coefficient	0.082	0.038	0.0303
Stall Angle	11 degrees	19	17

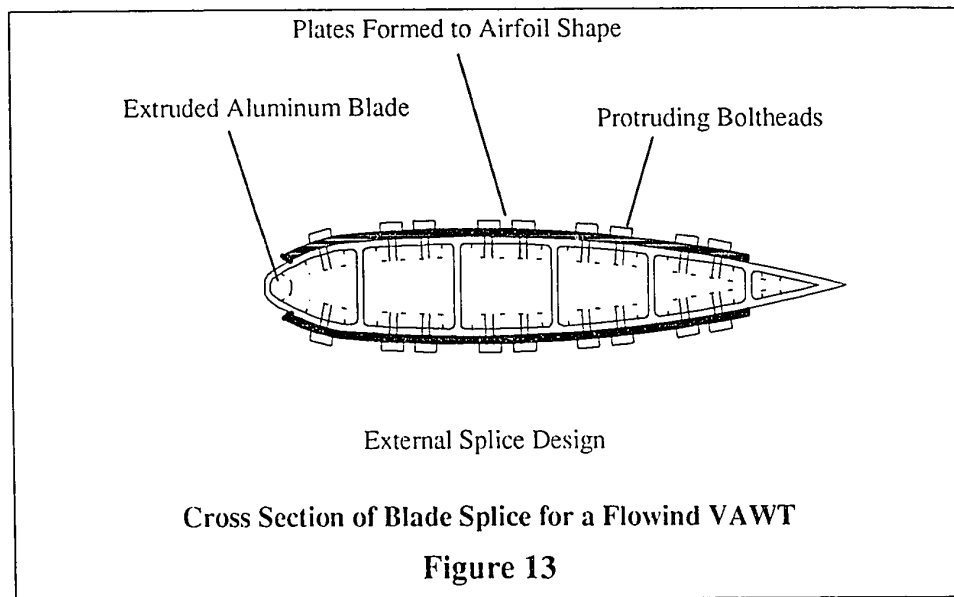
Table 2

Drag Data for a Strut Joint

Re = 1 million	Measured Data for Blade Joint	Measured Data for Faired Blade Joint	Original Eppler Data
Drag Coefficient	0.214	0.104	.0074
Angle-of-attack	0 degrees	0	0

Blade Patch Lift and Drag

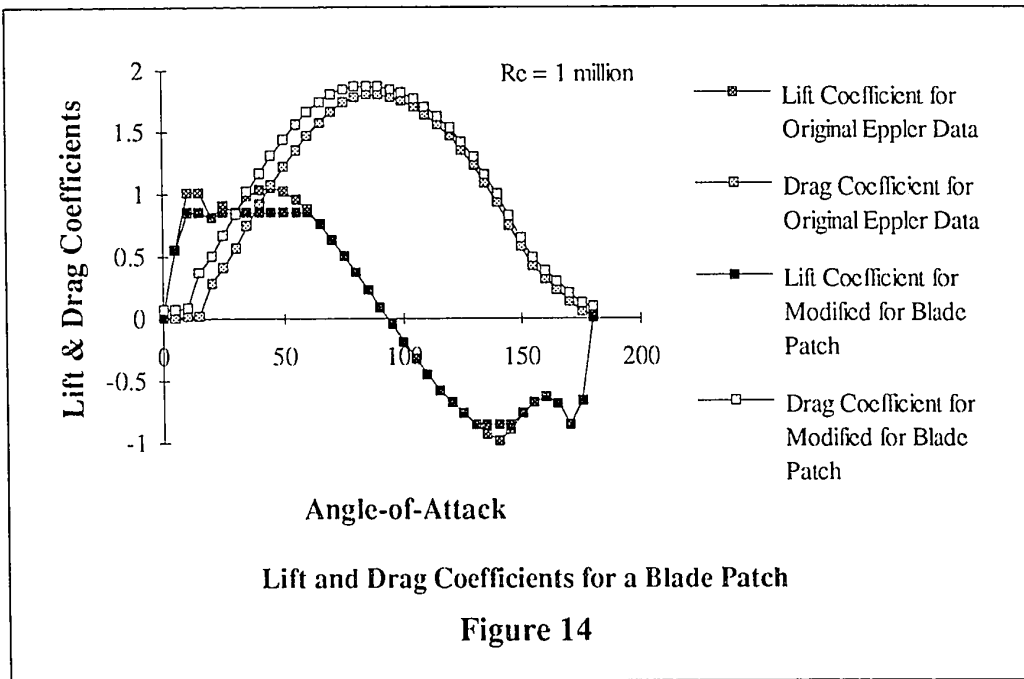
A blade patch is used to structurally reinforce a splice section of a Flowind F-19 turbine. As noted earlier the patch consists of two plates bolted to both sides of the turbine blade as shown in figure 13. The patch conforms to the NACA0015 shape but the plates do not cover the entire surface. The area around the leading edge remains uncovered. This leaves the sharp edges of the plates exposed near the front of the blade. Dozens of bolt heads protrude above the surface in a grid pattern, creating a rough surface.



The lift data for a patched splice is developed by adjusting the original Eppler data. The maximum value of the lift coefficient in the linear region of the curve, $C_l = 0.85$ at $\alpha = 8^\circ$, is set as an upper bound. For small angles-of-attack it is known that lift data is proportional to α . Even for a very rough patched surface it is reasonable to conclude that this will remain true for small angles-of-attack. Beyond the linear region, however, one cannot conclude that the lift will continue to rise as the original Eppler data does. The lift coefficient at the stall angle for an unpatched blade is also equal to 0.85. Decreasing the stall angle by adding a blade patch should produce a lift coefficient near this value at lower angles-of-attack. Note that this method is a conservative "best guess" at estimating the lift curve for a patched blade. Measured lift data would certainly provide more reliable results.

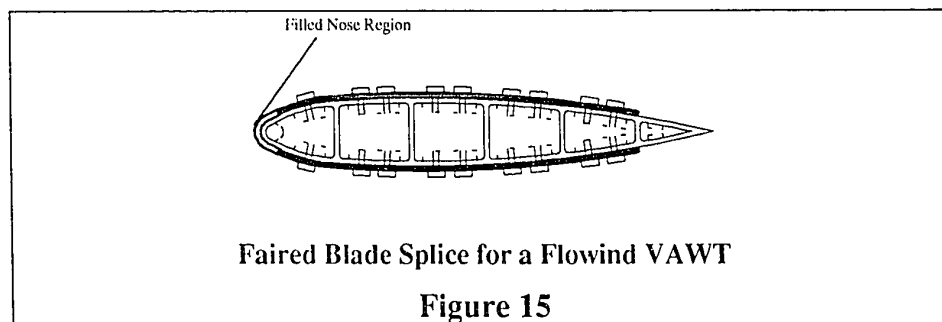
The drag data for a patched splice is also derived from the original Eppler data. The Eppler drag curve is shifted and increased to force it through the known drag data

measured by Westphal. For the patched blade the drag curve is *shifted* to the left by the change in the stall angle. The curve is then increased so that it intersects the measured drag coefficient below the stall angle. The results are shown in figure 14.



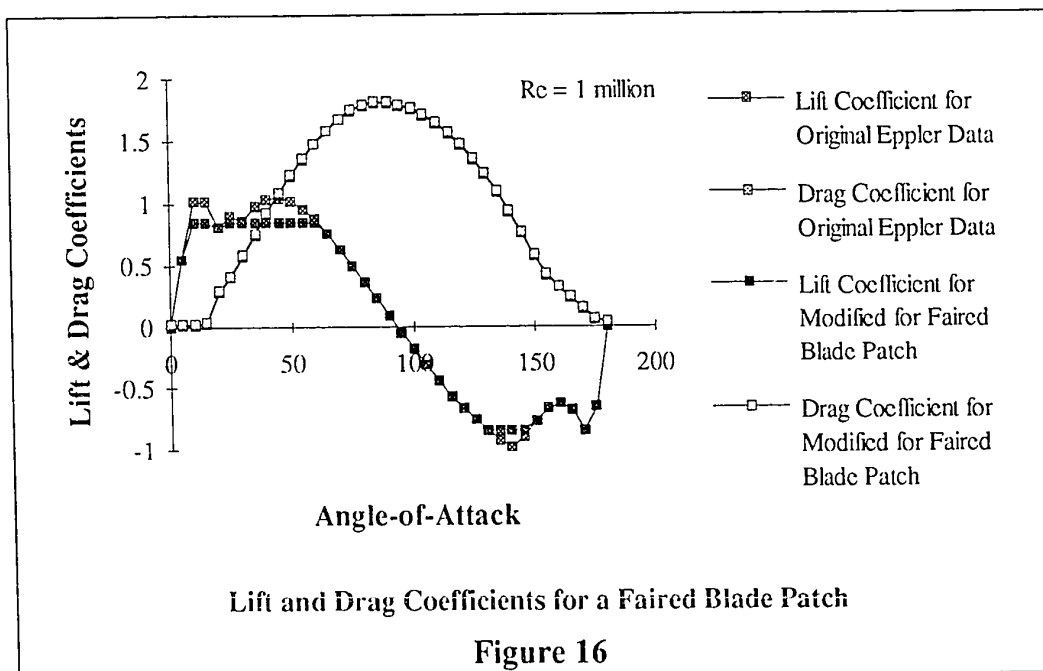
Faired Blade Patch Lift and Drag

A faired blade patch is essentially an ordinary bolted patch with a smoothed leading edge region. The nose region between the sharp edges of the plates is filled and formed to the NACA0015 shape as shown figure 15.



The lift data is produced in the same manner as the unfaired blade patch. The Eppler lift curve is truncated at the maximum linear region value.

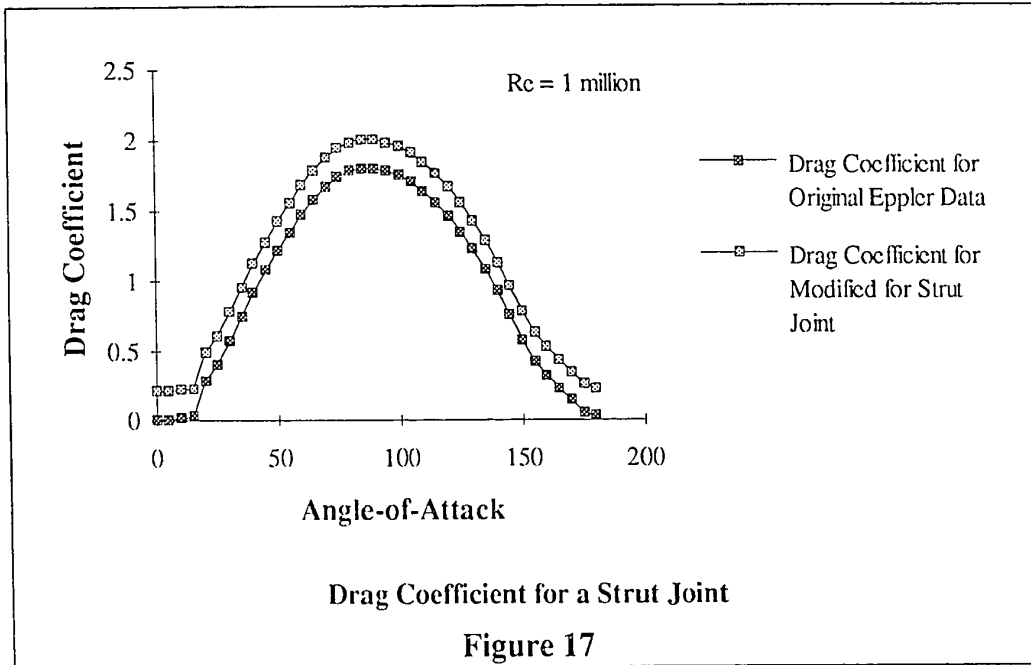
The drag curve is also developed as for the unfaired patch data. For the faired patch, however, there is no significant difference in the stall angle compared to the Eppler data so no leftward *shift* of the drag curve is necessary. The drag curve is only increased slightly to force it through the measured faired patch drag data as shown in figure 16.



Strut Joint Interference Lift and Drag Data

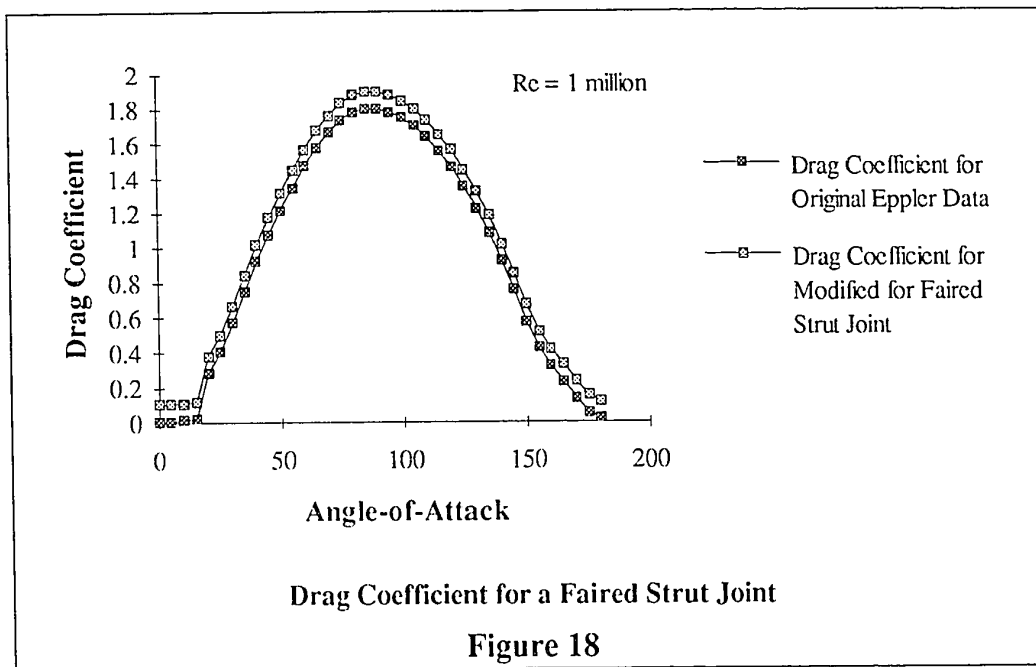
It is assumed that presence of the strut interfacing with the blade affects the local airflow so that essentially no lift will be produced in this region of the blade. The strut itself produces no lift and the strut joint is simply viewed as an extension of the strut. For this reason the lift on the blade segment near the joint is set equal to zero.

The drag is derived from the Eppler data and experimental measurements. There is no stall angle to account for so the drag curve is simply increased so that it contains known drag data for the local interference region around the joint. The resulting drag curve is shown in figure 17.



Faired Strut Joint Interference Lift and Drag Data

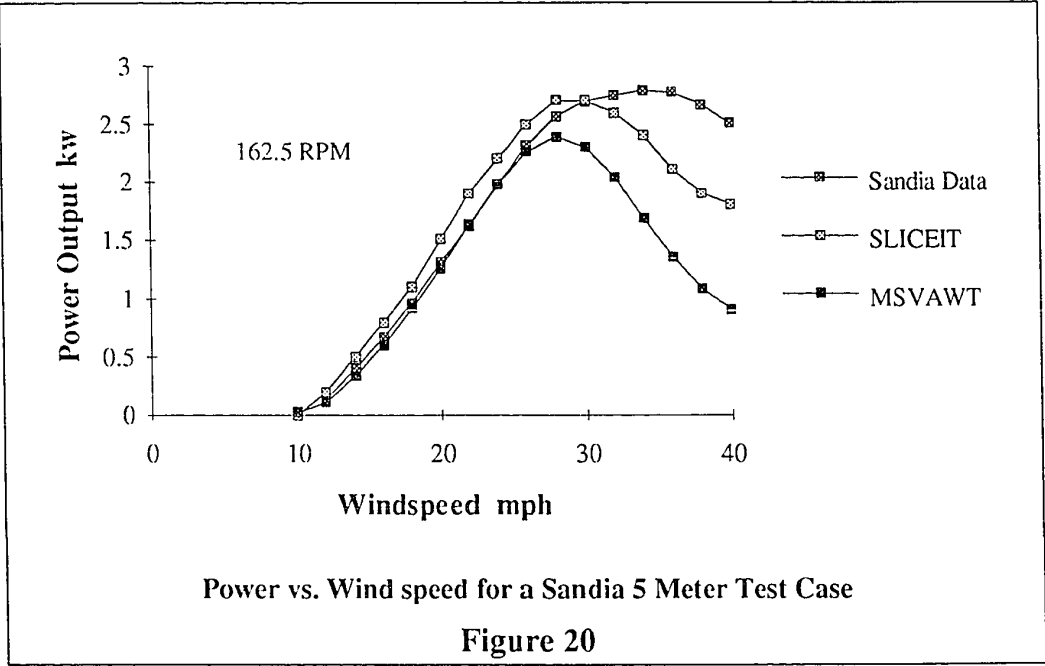
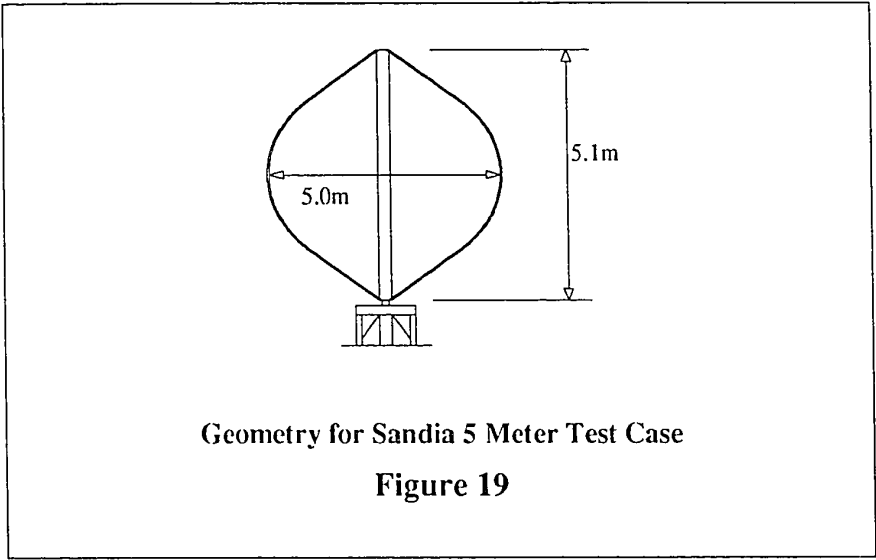
The faired strut joint consists of the original strut joint region with a bullet shaped fairing contoured around the strut and the blade. The lift is set to zero to produce a conservative estimate of the fairing influence. The drag curve is again developed from the original Eppler data as shown in figure 18. The Eppler drag curve is increased so that it intersects the measured drag data for a joint with a bullet fairing.



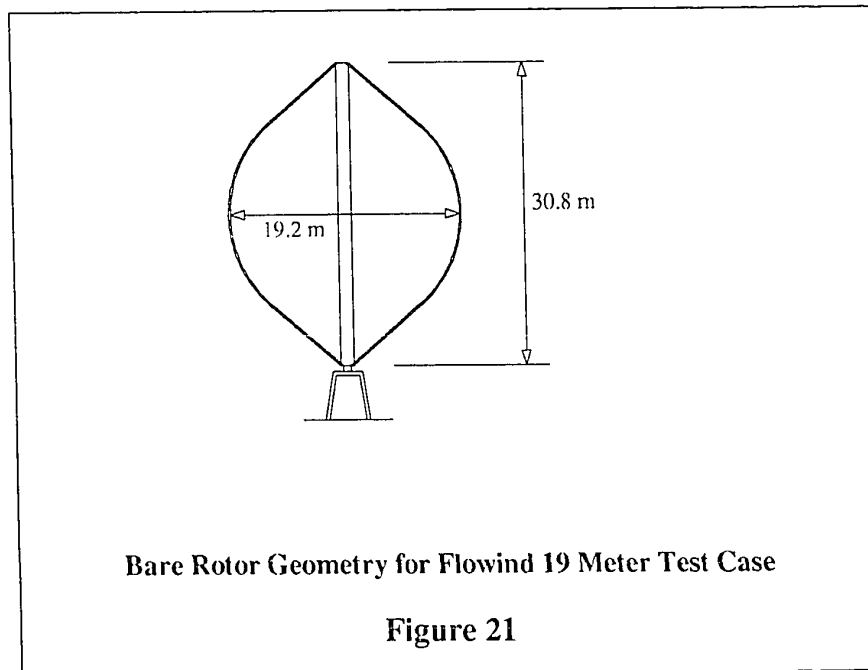
III TEST CASES

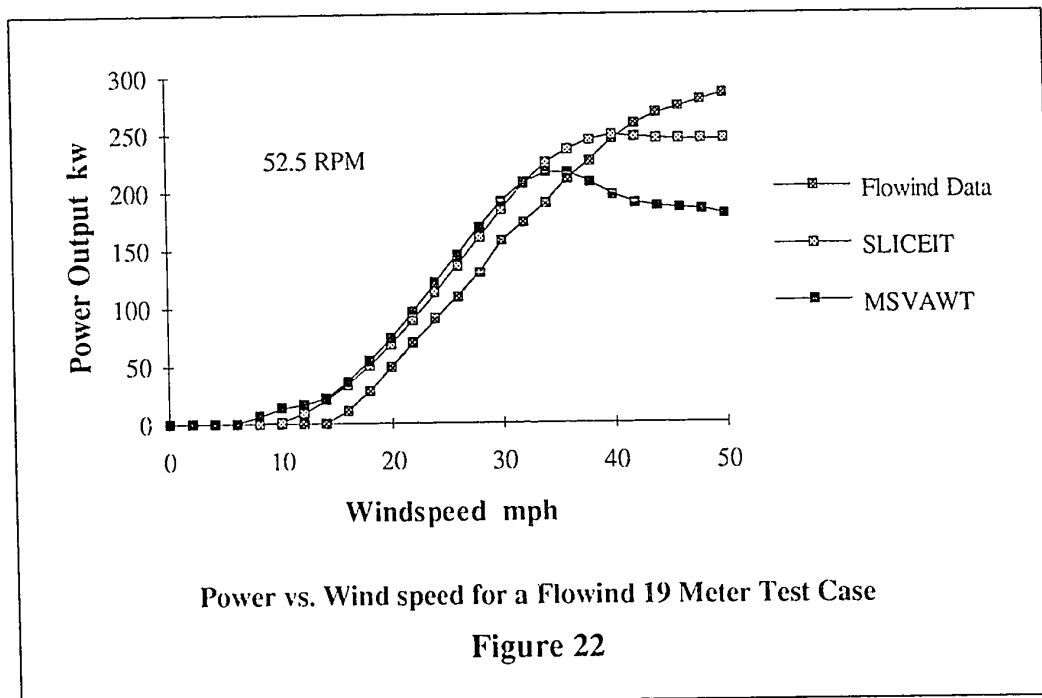
Two test cases are presented to compare output from the code used in this analysis to actual data and other accepted programs used in VAWT analysis. The first is for a Sandia 5 meter turbine. The geometry and power curves are shown in figures 19 & 20. The actual power curve measured at the Sandia test site and the power curve predicted by SLICEIT [22], an accepted computer program used to analyze VAWT performance, are plotted. The computer code used in this analysis, called MSVAWT for multiple streamtube vertical axis wind turbine, is presented for comparison. At lower wind speeds MSVAWT slightly under-predicts power output while at higher wind speeds it grossly under-predicts power. SLICEIT over-predicts power at lower wind speeds and does significantly better at predicting power output at the higher wind speeds. This is because SLICEIT employs a dynamic stall model which is crucial for estimating lift and drag coefficients at high wind speeds which result in large excursions in angle-of-attack.

Dynamic stall effects are known to become important at high wind speeds. The MSVAWT program, although it does not include a dynamic stall model, is still valid for the work done here because (i) only relative results compared to a nominal power curve are desired and (ii) most of the power produced by VAWTs occurs at modest wind speeds. The differences in annual energy production at higher wind speeds will not be significant compared to the bulk of the energy that is produced at lower wind conditions. Any multiple streamtube model will be equally affected by changes in lift and drag data. That is, the percent losses relative to the nominal power distribution will be essentially the same no matter what program is used.



The second test case is for a Flowind 19 meter turbine. Figures 21 & 22 show the geometry used and power curves as a function of wind speed. Again the actual power data and the SLICEIT results are presented for comparison to the MSVAWT results. MSVAWT over-predicts power output at lower wind speeds and under-predicts power at higher wind speeds. The "hump" that occurs at low wind speeds around 10 mph is due to convergence problems. When the streamtube velocity does not converge MSVAWT uses the freestream velocity to calculate the forces on the blade element. Because the freestream velocity is greater than the streamtube velocity, the forces on the blade and hence power output is over-predicted. SLICEIT does a slightly better job of predicting the power at lower wind speeds. At higher wind speeds SLICEIT does significantly better at estimating power than MSVAWT due to the dynamic stall model.





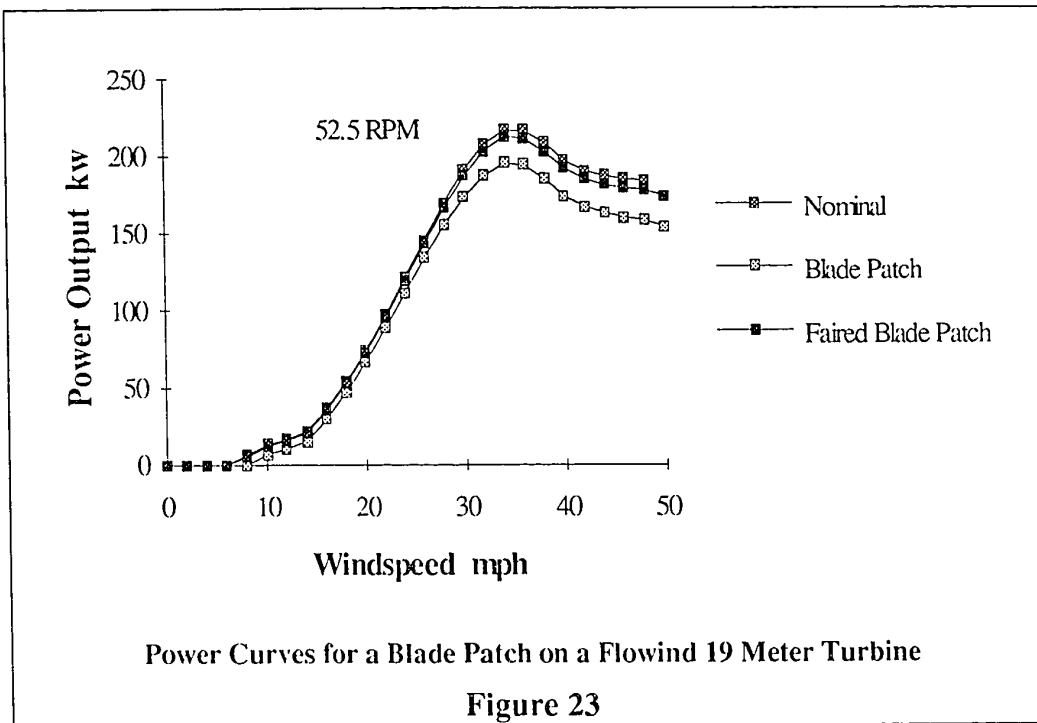
IV RESULTS

The MSVAWT program has been used to assess the effects of blade patches and strut joints on Flowind F-19 VAWTs. The power curves for a "bare" rotor without blade patches or strut joints were computed using Eppler lift and drag data. This established a nominal power curve. Modified Eppler data was then used to predict the power loss caused by blade patches and strut joints. The power curves for turbines with blade patches and strut joints were then compared to the nominal power curve. Annual energy production was computed using wind frequency distributions for two sites at the Flowind facility at the Altamont Pass. The annual energy production for turbines with patches and strut joints was then calculated as a percentage of the nominal energy production for one year.

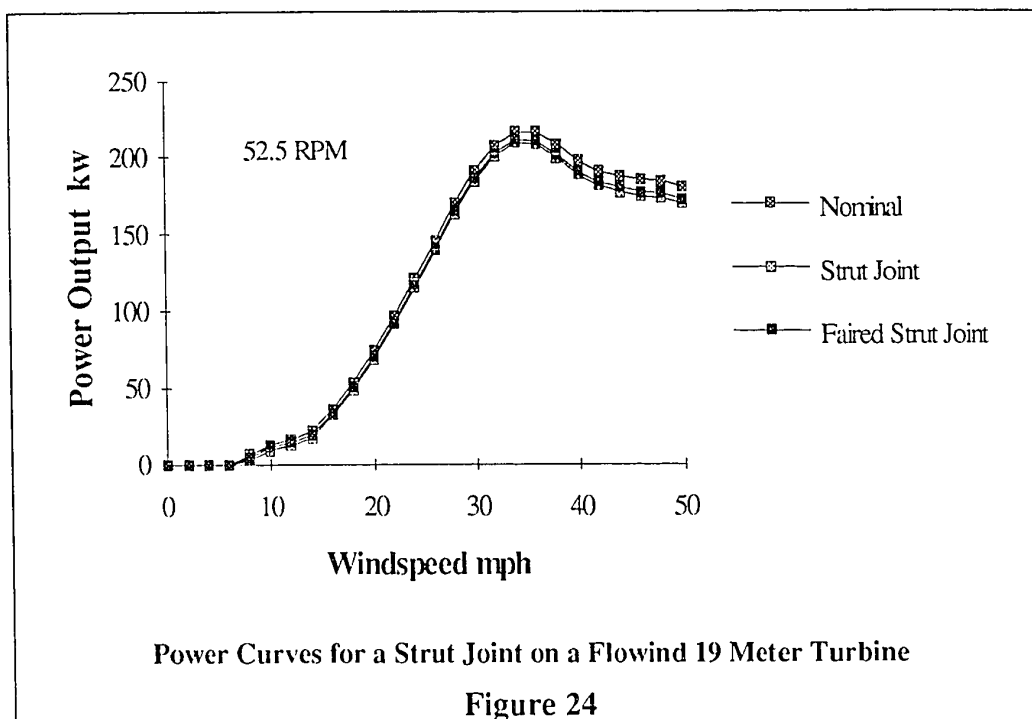
The two sites chosen for study represent the poorest and best wind conditions one could expect at the Altamont Pass facility. Site 1002 is characteristic of a poor case where most of the wind hours occur at low wind speeds from 5 to 20 mph. Site 1021 represents a distribution with the best expected wind conditions. The wind hours are spread across a range of wind speeds from 5 to 40 mph. Note that for a typical distribution the significant portion of power output occurs at moderate wind speeds below 30 mph. The poor power predictions of the MSVAWT program at higher wind speeds do not significantly affect the annual energy predictions because the contribution at wind speeds above 35 mph is minor.

The results of this work show the effects of faired and unfaired blade patches and strut joints on the power produced by Flowind F-19 vertical axis wind turbines. Two plots of power vs. wind speed are presented in figures 23 & 24. The first depicts the effects of

blade patches on an Flowind F-19 meter turbine. Clearly blade patches cause appreciable power loss compared to the nominal power curve that consists entirely of a NACA0015 airfoil section. Fairing the blade patches reduces power loss at all wind speeds.



The second plot shows the effects of strut joints on turbine power output. The strut joints produce noticeable power loss. Fairing the joint improves performance.



Integrating the product of the power curve and the wind distribution gives the total annual energy produced. The annual energy losses caused by the blade patches and strut joints are shown figure 25. The chart shows that blade patches and strut joints have appreciable impact on the annual energy output. For site 1002 blade patches alone reduce annual output by over 14%; the strut joint decreases energy output by over 8%. For site 1021 blade patches cause losses of over 11% and strut joints produce losses over 5%.

Fairing the patches and joints improves performance. Fairing a blade patch recovers about 12% of the total annual energy production for site 1002 and 9% for site 1021. This is mainly due to the increase in the stall angle. Fairing a patch also reduces overall drag and this too improves performance. Fairing a strut joint recovers about 2% of the annual

energy produced for both sites. This is because fairing a strut joint only decreases drag. No increases in performance occur due to improving the stall characteristics.

The combined effects of blade patches and strut joints are notable. For wind conditions at site 1002 about 22% of the annual energy produced is lost when both patches and joints are included. Approximately 14% can be recovered by using fairings. For site 1021 about 16% of the annual energy produced is lost; 11% can be recovered using fairings.

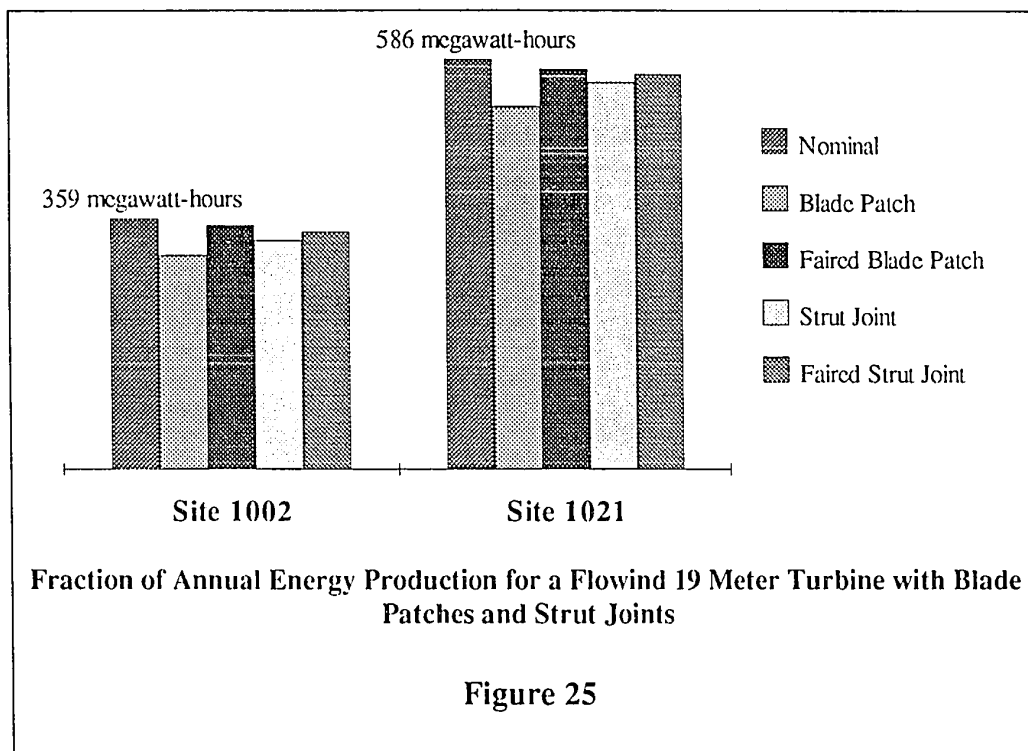


Table 3

Percent of Annual Energy Production for a Flowind 19 Meter Turbine with Blade Patches and Strut Joints

52.5 RPM	Blade Patch	Faired Blade Patch	Strut Joint	Faired Strut Joint
Site 1002 Nominal Power 359 megawatt-hours	85.4	97.3	91.3	94.8
Site 1021 Nominal Power 586 megawatt-hours	88.5	97.6	94.4	96.3

V CONCLUSIONS

The double multiple streamtube method has been implemented in a new computer code, MSVAWT, to estimate the power output and annual energy production of Flowind F-19 vertical axis wind turbines. Two test cases compared the power curves predicted by MSVAWT to power curves predicted by SLICEIT, a multiple streamtube program used in VAWT analysis, and actual power curves for a 5 meter & 19 meter turbine. At lower wind speeds there was good agreement with the SLICEIT program in both cases. At higher wind speeds, where dynamic stall becomes important, power estimates by MSVAWT were poor. Compared to measured power data MSVAWT accurately predicted turbine performance for the 5 meter test case at low wind speeds. For the 19 meter test case the power curve was over-predicted at low wind speeds. At higher wind speeds MSVAWT under-predicted the power curve for both test cases.

Combined with modified lift and drag data based on experimental results recently obtained at SJSU the MSVAWT program has been used to estimate of the effects of blade patches and strut joints on the power produced by vertical axis wind turbines. The stall angle was determined to be a key factor affecting power loss associated with blade patches. Increased drag caused by both the blade patches and strut joints significantly degrades turbine performance. Fairing both the blade patch and the strut joint will improve performance: about two thirds of the lost power production can be recovered by fairing both the patches and the joints.

Minimal lift and drag data for the structural components was required to assess the relative magnitudes of the power and energy production loss. Future modifications or

entirely new VAWT designs could be analyzed with this method. All that is needed is a few lift and drag coefficients at key Reynolds Numbers and angels-of-attack. Particularly important are changes in stall angles.

VI REFERENCES

- 1 Beedeel, Suzanne. *Windmills*, Charles Scribner's Sons, New York 1975
- 2 Eldridge, Frank R., *Wind Machines* 2nd edition, Van Nostrand Reinhold Company 1980.
- 3 "Wind Power", Mechanical Engineering Magazine, Sept. 1966
- 4 Torrey, V., *Wind-Catchers*, The Stephen Greene Press, Brattleboro, Vermont 1976.
- 5 Proceedings of the 1991 International VAWT Technology Workshop, April 11-13, 1991, Pleasonton, CA
- 6 Templin, R.J., "Aerodynamic Performance Theory for the NRC Vertical Axis Wind Turbine", NRC of Canada TR, LTR-LA-160, 1974.
- 7 Glauert, H., *The Elements of Aerofoil and Airscrew Theory*, 2nd Edition, Cambridge University Press, 1948.
- 8 Strickland, J.H., "The Darrieus Turbine: A Performance Prediction Model Using Multiple Streamtubes," Sandia Laboratories, Albuquerque, NM, Rept. SAND75-0431, Oct. 1975.
- 9 Paraschivoiu, I., "Aerodynamic Loads and Performance of the Darrieus Rotor," *Journal of Energy*, Vol. 6, Nov.-Dec. 1982, p. 406.
- 10 Paraschivoiu, I., "Double Multiple Streamtube Model with Recent Improvements," *Journal of Energy*, Vol. 7, May-June 1983, p. 250.
- 11 Parachivoiu, I., "Streamtube Expasion Effects on the Darrieus Wind Turbine," NASA Report 849051, 1984
- 12 Paraschivoiu, I., "Aerodynamic Analysis of the Darrieus Rotor Including Secondary Effects," *Journal of Energy*, Vol. 7, Sept.-Oct. 1983, p. 416.
- 13 Strickland, J.H., Webster, B.T. and Nguyen, T., "A Vortex Model of the Darrieus Turbine: An Analytical and Experimental Study," ASME Paper 79-WA/FE-6, 1979.
- 14 Rajagopalan, R. Ganesh, "Viscous Flow Field Analysis of a Vertical Axis Wind Turbine," NASA Report 869275, 1986

- 15 Gormont, R.E., "A Mathematical Model of Unsteady Aerodynamics and Radial Flow for Application to Helicopter Rotors," The Boeing Co., Vertol Division, Philadelphia, PA, USAAMRDL Tech. Rept. 72-67, May 1973.
- 16 Favier, D., Agnes, A., Barbi, C. and Maresca, C., "Combined Translation/Pitch Motion; A New Airfoil Dynamic Stall Simulation," Journal of Aircraft, Vol. 25, Sept. 1988, p. 805.
- 17 Berg, D.E. and Klimas, P.C., "VAWT Dynamic Stall Revisited," 12th Annual Energy Sources Technology, Houston, TX, Jan 22-25, 1989, p. 85.
- 18 Westphal, R.V., "VAWT Blade Patch Improvement," San Jose State University, San Jose, CA, SJSU Contract 24-1309-3462, 1991.
- 19 Westphal, R.V., Private Communication, Sept. 1992
- 20 Brown, A.B., *Fluid Mechanics of the Atmosphere*, Academic Press, Inc., San Diego, CA., 1991
- 21 Sheldahl, R.E., "Aerodynamic Characteristics of Seven Symmetrical Airfoil Sections Through 180-Degree Angle of Attack for use in Aerodynamic Analysis of Vertical Axis Wind Turbines," Sandia Laboratories, Albuquerque, NM, Rept. SAND80-2114, March 1981
- 22 Parachivoiu, I., "Double Multiple Streamtube Model for Darreius Wind Turbines," Second DOE-NASA Wind Turbine Dynamics Workshop, Feb 24-26, 1981, Cleveland, Ohio

Appendix A MSVAWT Computer Program Listing

The MSVAWT double multiple streamtube computer program models vertical axis wind turbine performance. Power output is predicted as function of wind speed. With a specified wind frequency distribution, annual energy production is calculated.

The program was designed to flexibly handle a wide variety of cases. Any set of wind-hours can be specified to represent the conditions that occur at a given site. The rotor geometry is discretized and divided into sections. This allows any shape or design to be studied. Lift and drag data can be input according to available data. If only limited experimental data is available the program interpolates between the listed values of Reynolds numbers and angles of attack. When more complete data is input, such as the Eppler data for NACA airfoils, the program again interpolates to find lift and drag coefficients; in this case, though, the coefficients will be confined to a smaller, more accurate range.

Input

Input to MSVAWT is divided into three categories each forming a separate input file:

1. Wind parameters and the wind-hours distribution *windfile*
2. Lift and drag data *clcdfile*
3. Geometry data *geomfile*

One additional file, the *startfile*, is required to run the program. When the program is executed it prompts the user for a *startfile*. Included in the *startfile* are the three input files - one for each of the categories mentioned. This specifies the wind hours distribution, lift & drag data and the geometry to be studied. By editing the *startfile* various combinations of wind conditions, lift & drag data, geometry and rotor rotation speed can be easily studied. Rarely will the *clcdfile* need to be changed. Arrays of lift and drag for several airfoil sections can be accommodated in one file. Depending on memory limits specific to compilers and computers, however, this file may need to be modified if it becomes too large.

Format for the Input Files:

```
startfile
  windfile
  clcdfile
  geomfile
  rotor speed RPM
```

windfile
number wind speeds n, fric height m, gradient height m, $r \text{ kg/m}^3$, $m \text{ kg/(m}^2\text{sec)}$
wind speed 1 m/s, number of wind-hours at wind speed 1
wind speed 2 m/s, number of wind-hours at wind speed 2
wind speed 3 m/s, number of wind-hours at wind speed 3
...
wind speed n m/s, number of wind-hours at wind speed n

clcdfile

number of airfoil data sets an
number of Re 1 Rn, Re max 1, Re min 1, airfoil thickness 1 m
Re 1, number of angles of attack 1 n
alpha 1, Cl 1, Cd 1
alpha 2, Cl 1, Cd 2
alpha 3, Cl 1, Cd 3
...
alpha n, Cl n, Cd n
Re 2, number of angles of attack 2 n
alpha 1, Cl 1, Cd 1
alpha 2, Cl 1, Cd 2
alpha 3, Cl 1, Cd 3
...
alpha n, Cl n, Cd n
...
Re Rn, number of angles of attack Rn n
alpha 1, Cl 1, Cd 1
alpha 2, Cl 1, Cd 2
alpha 3, Cl 1, Cd 3
...
alpha n, Cl n, Cd n
number of Re 2 Rn, Re max 2, Re min 2, airfoil thickness 2 m
Re 1, number of angles of attack n
alpha 1, Cl 1, Cd 1
alpha 2, Cl 1, Cd 2
alpha 3, Cl 3, Cd 3
...
alpha n, Cl n, Cd n

Re 2, number of angles of attack 2

...

Re Rn, number of angles of attack m

...

number of Re an Rn, Re max an, Re min an, airfoil thickness an m

Geomfile

number of sections sn, z start height m
section height 1 m, number of z divisions 1 nz, number of theta div 1 nt
radius 1 m, cord length 1 m, length of element 1 m, airfoil section 1
radius 2 m, cord length 2 m, length of element 2 m, airfoil section 2
radius 3 m, cord length 3 m, length of element 3 m, airfoil section 3
...
radius nz m, cord length nz m, length of element nz m, airfoil sec nz
section height 2 m, number of z divisions 2 nz, number of theta div 2 nt
radius 1 m, cord length 1 m, length of element 1 m, airfoil section 1
radius 2 m, cord length 2 m, length of element 2 m, airfoil section 2
radius 3 m, cord length 3 m, length of element 3 m, airfoil section 3
...
radius nz m, cord length nz m, length of element nz m, airfoil sec nz
...
section height sn m, number of z divisions sn nz, number of theta div sn nt
radius 1 m, cord length 1 m, length of element 1 m, airfoil section 1
radius 2 m, cord length 2 m, length of element 2 m, airfoil section 2
radius 3 m, cord length 3 m, length of element 3 m, airfoil section 3
...
radius nz m, cord length nz m, length of element nz m, airfoil sec nz

Outputs

Output from MSVAWT can be printed to the screen at any point in the program by setting certain print flags within the code before compiling. The output can easily be redirected to files for analysis with spreadsheets. The most common output is annual energy production and the power vs. wind speed data. Input data can be echoed for checking purposes. Output from deeper loops within the program progressively produces more and more data. Forces, velocity components, lift and drag coefficients and convergence parameters can be printed from nested loops within the program.

PRINTOUT

```

#include <stdio.h>
#include <math.h>

#define PI 3.1415927 /* radians */
#define z10 10.0 /* meters */
#define von_karman 0.4 /* dimensionless */
#define N_wind 60
#define N_Re 12
#define N_alpha 26
#define N_theta 37
#define N_z 22
#define N_clcd 5
#define N_sec 10
#define N_char 13
#define N_blades 2
/*=====*/
float W(z,vl0,zo)
{
    float z,vl0,zo;
    float u_fric;
    u_fric = vl0*von_karman/log((z10+zo)/zo);
    return(u_fric/von_karman*log((z+zo)/zo));
}
/*=====*/

main()
{
    FILE *start_file,*wind_file,*clcd_file,*geom_file;
    char startfile[N_char],windfile[N_char],clcdfile[N_char],geomfile[N_char];
    char clcdtitle[72];
    float wind[N_wind],windhours[N_wind],windin[N_wind];
    float height[N_sec],zo,grad_H,rho,mu,u_fric;

```

```

float radius[N_sec][N_z],dlength[N_sec][N_z];
float cord[N_sec][N_z];
int z_div[N_sec],zdiv,theta_div[N_sec];
int num_wind,num_sec,num_ClCd,ClCdnum[N_sec][N_z],clcdnum;
int a,b,c,e,f,g,h,i,j,k,l,m,n;
float dz,dtheta,T,T_sec,z,r,w,s,T_z,theta,v,vx,vy,Re;
float alpha,alpha_Ldeg,alpha_Ddeg;
float v2,alpha_L,alpha_D,alpha_pos;
float cl,cl1,cl2,cd,cd1,cd2,plan_area,fl,fd,T_theta;
float CL[N_clcd][N_Re][N_alpha],CD[N_clcd][N_Re][N_alpha];
float RE[N_clcd][N_Re],ALPHA[N_clcd][N_Re][N_alpha];
float Re_min[N_clcd],Re_max[N_clcd];
int n_alpha[N_clcd][N_Re],n_Re[N_clcd];
float z_start,omega,RPM,tc[N_clcd];
int wind_flag1,wind_flag2,start_flag,clcd_flag1,clcd_flag2;
int geom_flag1,geom_flag2,I_flag,II_flag,III_flag,IV_flag1,IV_flag2;
int dyn_flag1,dyn_flag2;
float u,uu[N_theta],au,ft,fn,fx,fxi,fi,fj,aj;
int au_flag,it_flag,V_flag;
int itterate,itcount;
float udif,ulast;
float power,power_hours;

/*===== DEFINE PRINT FLAGS =====*/
wind_flag1 = 1; /* windfile,num_wind,zo,grad_H,rho,mu */
wind_flag2 = 1; /* wind[a],windhours[a] */
start_flag = 1; /* startfile,RPM,omega */
clcd_flag1 = 1; /* clcdfile,n_Re,Re_min,Re_max,tc */
clcd_flag2 = 0; /* RE[n][a],ALPHA[n][a][b],CL[n][a][b],CD[n][a][b] */
geom_flag1 = 1; /* geomfile,num_start,z_start,height,z_div,theta_div */
geom_flag2 = 0; /* radius[n][a],cord[n][a],dlength[n][a],ClCdnum[n][a] */
I_flag = 1; /* wind[g],windhours[g],T,power */
II_flag = 0; /* section # (h), Tsec, dtheta,dz */

```

```

III_flag = 0; /* z,T_z,r,w,u,s,dlenght[i],cord[i],plan_area,fi,ai */
IV_flag1 = 0; /* theta,v,Re,alpha_deg,T_theta */
IV_flag2 = 0; /* w,vx,vy,cl,cd,fl,fd,ft,fn,fx */
au_flag = 0; /* w,u,au,ulast,udif,fi */
it_flag = 0; /* print message for a failed convergence */
/*===== OPEN START FILE =====*/
printf("\n Enter start filename ");
scanf("%s",&startfile);
start_file = fopen(startfile,"r");
if(start_file==NULL){printf("\n\n CANNOT OPEN START FILE \n\n"); exit(1);}
fscanf(start_file,"%s",&windfile);
fscanf(start_file,"%s",&clcdfile);
fscanf(start_file,"%s",&geomfile);
fscanf(start_file,"%f",&RPM);
omega = RPM*2.0*PI/60.0;
fclose(start_file);

if(start_flag==1)
{
    printf("\n startfile= %s, windfile= %s, clcdfile= %s, geomfile= %s",
        startfile,windfile,clcdfile,geomfile);
    printf("\n RPM= %f, omega= %f",RPM,omega);
}
/*===== GET INPUT FROM WIND FILE =====*/
wind_file = fopen(windfile,"r");
if(wind_file==NULL){printf("\n\n CANNOT OPEN WIND FILE \n\n"); exit(1);}
fscanf(wind_file,"%d",&num_wind);
fscanf(wind_file,"%f",&z0);
fscanf(wind_file,"%f",&grad_H);
fscanf(wind_file,"%f",&rho);
fscanf(wind_file,"%f",&mu);

if(wind_flag1==1)

```

```

printf("\n windfile= %s, num_wind= %d, zo= %f, grad_H= %f, rho= %f, mu= %f",
      windfile,num_wind,zo,grad_H,rho,mu);

for (a=1;a<=num_wind;++a)
{
    fscanf(wind_file,"%f",&wind[a]);
    fscanf(wind_file,"%f",&windhours[a]);

    if(wind_flag2==1)
        printf("\n wind[a]= %f, windhours[a]= %f",wind[a],windhours[a]);
}
fclose(wind_file);
/*===== GET CLCD DATA =====*/
clcd_file = fopen(clcdfile,"r");
if(clcd_file==NULL){printf("\n\n CANNOT OPEN CLCD FILE \n\n"); exit(1);}
fscanf(clcd_file,"%d",&num_CLCD);
for(n=1;n<=num_CLCD;++n)
{
    fscanf(clcd_file,"%s",clcdtitle);
    fscanf(clcd_file,"%d",&n_Re[n]);
    fscanf(clcd_file,"%f",&Re_min[n]);
    fscanf(clcd_file,"%f",&Re_max[n]);
    fscanf(clcd_file,"%f",&tc[n]);

    if(clcd_flag1==1)
    {
        printf("\n clcdtitle= %s, num_CLCD= %d",clcdtitle,num_CLCD);
        printf("\n n_Re[n]= %d, Re_min[n]= %f, Re_max[n]= %f, tc[n]= %f",
              n_Re[n],Re_min[n],Re_max[n],tc[n]);
    }
    for(a=1;a<=n_Re[n;++a)
    {
        fscanf(clcd_file,"%f",&RE[n][a]);

```

```

fscanf(clcd_file, "%d", &n_alpha[n][a]);
for(b=1; b<=n_alpha[n][a]; ++b)
{
    fscanf(clcd_file, "%f", &ALPHA[n][a][b]);
    fscanf(clcd_file, "%f", &CL[n][a][b]);
    fscanf(clcd_file, "%f", &CD[n][a][b]);

    if(clcd_flag2==1)
    {
        printf("\n      RE[n][a]= %f, ALPHA[n][a][b]= %f",
              RE[n][a], ALPHA[n][a][b]);
        printf("\n      CL[n][a][b]= %f, CD[n][a][b]= %f",
              CL[n][a][b], CD[n][a][b]);
    }
}
}
fclose(clcd_file);
/*===== READ IN GEOMETRY DATA FOR SECTION =====*/
geom_file = fopen(geomfile, "r");
if(geom_file==NULL){printf("\n\n CANNOT OPEN GEOMETRY FILE \n\n"); exit(1);}
fscanf(geom_file, "%d", &num_sec);
fscanf(geom_file, "%f", &z_start);
for(n=1; n<=num_sec; ++n)
{
    fscanf(geom_file, "%f", &height[n]);
    fscanf(geom_file, "%d", &z_div[n]);
    fscanf(geom_file, "%d", &theta_div[n]);

    if(geom_flag1==1)
    {
        printf("\n geomfile= %s, num_sec= %d, z_start= %f, height[n]= %f",
              geomfile, num_sec, z_start, height[n]);
    }
}

```

```

    printf("\n z_div[n]= %d, theta_div[n]= %d",
           z_div[n],theta_div[n]);
}
zdiv = z_div[n]; if(zdiv<0) zdiv=-zdiv;
for(a=1;a<=zdiv;++a)
{
    fscanf(ggeom_file,"%f",&radius[n][a]);
    fscanf(ggeom_file,"%f",&cord[n][a]);
    fscanf(ggeom_file,"%f",&dlength[n][a]);
    fscanf(ggeom_file,"%d",&ClCdnm[n][a]);

    if(ggeom_flag2==1)
    {
        printf("\n radius[n][a]= %f, cord[n][a]= %f",
               radius[n][a],cord[n][a]);
        printf("\n dlength[n][a]= %f, ClCdnm[n][a]= %d",
               dlength[n][a],ClCdnm[n][a]);
    }
}
fclose(ggeom_file);
/*===== START CALC FOR A WIND SPEED =====*/
/*===== power_hours = 0.0; =====*/
for(g=1;g<=num_wind;++g)
{
    /*===== START TORQUE CALC =====*/
    T = 0.0;
    z = z_start;

    for(h=1;h<=num_sec;++h)
    {

```



```

/*===== START CALC T_sec =====*/

dtheta = 2.0*PI/theta_div[h];
dz = height[h]/z_div[h];
T_sec = 0.0;

for(i=1;i<=z_div[h];++i)
{
/*===== CALC T_Z'S ON SECTION =====*/
plan_area = dlength[h][i]*cord[h][i];
z = z + dz/2.0;
w = W(z,wind[g],zo);
z = z + dz/2.0;
s = radius[h][i]*omega;
theta = dtheta/2;

T_z = 0.0;
for(j=1;j<=theta_div[h]/2;++j)
{
/*===== CALC T_THETA'S ON UPWIND SIDE OF ROTOR =====*/
aj = radius[h][i]*dtheta*dz*sin(theta); if(aj<0.0) aj= -aj;
ulast = w;
au = 0.0;
u = w*(1.0-au);
itcount = 0;
ittrate = 1;
while(ittrate==1)
{
/*===== ITERATE TO FIND fx =====*/
if(u<0.0) {u=w; ittrate=0; if(it_flag==1)
printf("\n &&& u less than zero &&& +j= %d",j);}
if(itcount==10) {u=w; ittrate=0; if(it_flag==1)
printf("\n !!!! ITCOUNT MAX !!!! +j= %d",j);}
}
}
}

```

```

vx = u*cos(theta)+s;
vy = u*sin(theta)*dz/dlength[h][i];
v2 = vx*vx+vy*vy;
v = sqrt(v2);
if (vx==0.0&&vy>=0.0) alpha = PI/2;
else if (vx==0.0&&vy<0.0) alpha = -PI/2;
else
{
    alpha = atan(vy/vx);
    if (vx<0.0&&vy>0.0) alpha = alpha+PI;
    if (vx<0.0&&vy<0.0) alpha = alpha-PI;
}
alpha_pos = alpha; if (alpha_pos<0.0) alpha_pos = -alpha_pos;
alpha_L = alpha;
alpha_D = alpha;
alpha_Ldeg = alpha_L*180.0/PI;
if (alpha_Ldeg<0.0) alpha_Ldeg = -alpha_Ldeg;
alpha_Ddeg = alpha_D*180.0/PI;
if (alpha_Ddeg<0.0) alpha_Ddeg = -alpha_Ddeg;

Re = rho*v*cord[h][i]/mu;

clcdnum = Clcdnum[h][i];
if (Re<Re_min[clcdnum]) Re = Re_min[clcdnum];
if (Re>Re_max[clcdnum]) Re = Re_max[clcdnum];
k = 1; while (Re[clcdnum][k+1]<Re) k = k+1;
l = 1; while (ALPHA[clcdnum][k][l+1]<alpha_Ldeg) l = l+1;
m = 1; while (ALPHA[clcdnum][k][m+1]<alpha_Ddeg) m = m+1;
c11 = CL[clcdnum][k][l] + (alpha_Ldeg-ALPHA[clcdnum][k][l])/
    (CL[clcdnum][k][l+1]-CL[clcdnum][k][l]);
    (ALPHA[clcdnum][k][l+1]-ALPHA[clcdnum][k][l]);
c12 = CL[clcdnum][k+1][l] + (alpha_Ldeg-ALPHA[clcdnum][k+1][l])*

```

```

(CL[clcdnum][k+1][l+1]-CL[clcdnum][k+1][l])/
(ALPHA[clcdnum][k+1][l+1]-ALPHA[clcdnum][k+1][l]);
cl = cl1 + (Re-RE[clcdnum][k])* (cl2-cl1)/
(RE[clcdnum][k+1]-RE[clcdnum][k]);
cd1 = CD[clcdnum][k][m] + (alpha_Ddeg-ALPHA[clcdnum][k][m])*
(CD[clcdnum][k][m+1]-CD[clcdnum][k][m])/
(ALPHA[clcdnum][k+1][m] + (alpha_Ddeg-ALPHA[clcdnum][k+1][m])*
(CD[clcdnum][k+1][m+1]-CD[clcdnum][k+1][m])/
(ALPHA[clcdnum][k+1][m+1]-ALPHA[clcdnum][k+1][m]);
cd = cd1 + (Re-RE[clcdnum][k])* (cd2-cd1)/
(RE[clcdnum][k+1]-RE[clcdnum][k]);

fl = cl*rho*v*v*plan_area/2; if (alpha<0.0) fl=-fl;
fd = cd*rho*v*v*plan_area/2;
ft = fl*sin(alpha)-fd*cos(alpha);
fn = fl*cos(alpha)+fd*sin(alpha);
fx = fn*sin(theta)*dz/dlength[h][i]+ft*cos(theta);

fj = N_blades*fx*dtheta/(2.0*PI);
au = au*au+fj/(2.0*rho*aj*w*w);
u = w*(1.0-au);
udif = ulast-u; if (udif/ulast<0.0) udif= -udif;
if (udif/ulast<0.01) itterate = 0;

if (au_flag==1)
{
printf("\n j=%2d theta=%6.3f w= %6.3f u= %6.3f au=%6.3f",
j,theta,w,u,au);
printf("\n ulast= %6.3f udif=%10.6f fj= %10.6f aj=%10.6f",
ulast,udif,fj,aj);
}
ulast = u;

```

```

        itcount = itcount+1;
    } /* end first while */
    uu[j] = 2.0*u-w;

    T_theta = N_blades*radius[h][i]*ft;
    T_z = T_z + T_theta;

    if (IV_flag1==1&&j==1)
        printf("\n theta,v,Re,alpha,alpha_deg,T_theta");
    if (IV_flag2==1&&j==1)
        printf("w,vx,vy,cl,cd,fl,fd,ft,fn,fx");
    if (IV_flag1==1)
        printf("\n %f,%f,%f,%f,%f,%f",
                theta,v,Re,alpha,sqrt(alpha*alpha)*180.0/PI,T_theta);
    if (IV_flag2==1)
        printf(" %f,%f,%f,%f,%f,%f,%f,%f,%f",
                w,vx,vy,cl,cd,fl,fd,ft,fn,fx);

    theta = theta+dtheta;
} /* end first j loop */

for (j=theta_div[h]/2; j>=1; j--)
{
    /*===== CALC T_THETA'S ON DOWNWIND SIDE OF ROTOR =====*/
    aj = radius[h][i]*dtheta*dz*sin(theta); if (aj<0.0) aj= -aj;
    ulast = uu[j];
    au = 0.0;
    u = uu[j]*(1.0-au);
    itcount = 0;
    itterate = 1;
    while (itterate==1)
    {
        if (u<0.0) {u=w; itterate=0;

```

```

        if(it_flag==1)
            printf("\n &&&& u less than zero &&&& -j= %d", j);}
    if(itcount==10) {u=w; itterate=0;
        if(it_flag==1)
            printf("\n !!!!! ITCOUNT MAX !!!!! -j= %d", j);}

    vx = u*cos(theta)+s;
    vy = u*sin(theta)*dz/dlength[h][i];
    v2 = vx*vx+vy*vy;
    v = sqrt(v2);
    if(vx==0.0&&vy>=0.0) alpha = PI/2;
    else if(vx==0.0&&vy<0.0) alpha = -PI/2;
    else
    {
        alpha = atan(vy/vx);
        if(vx<0.0&&vy>0.0) alpha = alpha+PI;
        if(vx<0.0&&vy<0.0) alpha = alpha-PI;
    }
    alpha_pos = alpha; if(alpha_pos<0.0) alpha_pos = -alpha_pos;
    alpha_L = alpha;
    alpha_D = alpha;
    alpha_Ldeg = alpha_L*180.0/PI;
    if(alpha_Ldeg<0.0) alpha_Ldeg = -alpha_Ldeg;
    alpha_Ddeg = alpha_D*180.0/PI;
    if(alpha_Ddeg<0.0) alpha_Ddeg = -alpha_Ddeg;

    Re = rho*v*cord[h][i]/mu;

    clcdnum = ClCdnun[h][i];
    if(Re<Re_min[clcdnum]) Re = Re_min[clcdnum];
    if(Re>Re_max[clcdnum]) Re = Re_max[clcdnum];
    k = 1; while(Re[clcdnum][k+1]<Re) k = k+1;
    l = 1; while(ALPHA[clcdnum][k][l+1]<alpha_Ldeg) l = l+1;

```

```

m = 1; while (ALPHA[clcdnum][k][m+1]<alpha_Ddeg) m = m+1;
cl1 = CL[clcdnum][k][1] + (alpha_Ldeg-ALPHA[clcdnum][k][1]) *
      (CL[clcdnum][k][1+1]-CL[clcdnum][k][1])/
      (ALPHA[clcdnum][k][1+1]-ALPHA[clcdnum][k][1]);
cl2 = CL[clcdnum][k+1][1] + (alpha_Ldeg-ALPHA[clcdnum][k+1][1]) *
      (CL[clcdnum][k+1][1+1]-CL[clcdnum][k+1][1])/
      (ALPHA[clcdnum][k+1][1+1]-ALPHA[clcdnum][k+1][1]);
cl = cl1 + (Re-RE[clcdnum][k]) * (cl2-cl1) /
      (RE[clcdnum][k+1]-RE[clcdnum][k]);
cd1 = CD[clcdnum][k][m] + (alpha_Ddeg-ALPHA[clcdnum][k][m]) *
      (CD[clcdnum][k][m+1]-CD[clcdnum][k][m])/
      (ALPHA[clcdnum][k][m+1]-ALPHA[clcdnum][k][m]);
cd2 = CD[clcdnum][k+1][m] + (alpha_Ddeg-ALPHA[clcdnum][k+1][m]) *
      (CD[clcdnum][k+1][m+1]-CD[clcdnum][k+1][m])/
      (ALPHA[clcdnum][k+1][m+1]-ALPHA[clcdnum][k+1][m]);
cd = cd1 + (Re-RE[clcdnum][k]) * (cd2-cd1) /
      (RE[clcdnum][k+1]-RE[clcdnum][k]);

fl = cl*rho*v*v*plan_area/2; if (alpha<0.0) fl=-fl;
fd = cd*rho*v*v*plan_area/2;
ft = fl*sin(alpha)-fd*cos(alpha);
fn = fl*cos(alpha)+fd*sin(alpha);
fx = fn*sin(theta)*dz/dlength[h][i]+ft*cos(theta);

fj = N_blades*f*x*dtheta/(2.0*PI);
au = au*au+fj/(2.0*rho*aj*uu[j]*uu[j]);
u = uu[j]*(1.0-au);
udif = ulast-u; if (udif<0.0) udif = -udif;
if (udif/ulast<0.01) itterate = 0;

if (au_flag==1)
{
printf("\n j=%2d theta=%6.3f uu[j]=%6.3f u=%6.3f au=%6.3f",

```

```

        j,theta,uu[j],u,au);
        printf("\n ulast= %6.3f udif=%10.6f fj=%10.6f aj=%10.6f",
            ulast,udif,fj,aj);
    }
    ulast = u;
    itcount = itcount+1;
} /* end second while */

T_theta = N_blades*radius[h][i]*ft;
T_z = T_z + T_theta;

if(IV_flag1==1&&j==1)
    printf("\n theta,v,Re,alpha,alpha_deg,T_theta");
if(IV_flag2==1&&j==1)
    printf("w,vx,vy,cl,cd,fl,fd,ft,fn,fx");
if(IV_flag1==1)
    printf("\n %f,%f,%f,%f,%f",
        theta,v,Re,alpha,sqrt(alpha*alpha)*180.0/PI,T_theta);
if(IV_flag2==1)
    printf(" %f,%f,%f,%f,%f,%f,%f,%f",
        w,vx,vy,cl,cd,fl,fd,ft,fn,fx);

    theta = theta+dtheta;
} /* end second j loop */

T_z = T_z/theta_div[h];
T_sec = T_sec + T_z;

if(III_flag==1)
{
    printf("\n
        z= %f, T_z= %f, r= %f, w= %f, u= %f, s= %f",
        z-dz/2,T_z,radius[h][i],w,u,s);
    printf("\n
        dlength[h][i]= %f, cord[h][i]= %f, plan_area= %f",

```

```

        dlength[h][i],cord[h][i],plan_area);
    }
}
T = T + T_sec;

if(II_flag==1)
    printf("\n Section # = %3d, Tsec= %.2e",h,T_sec);
}

power = T*omega;
if(power>0.0) power_hours = power_hours+power*windhours[g];
if(II_flag==1)
    printf("\n w[g]= %5.1f, wh[g]= %5.1f, T= %.2e, P= %.2e",
           wind[g],windhours[g],T,power);
}
printf("\n power_hours= %f",power_hours);
/*===== THIS IS THE END =====*/
printf("\n\n");
return(0);
}•

```


Appendix B Sandia 5 Meter Test Case Input

Input file to SLICEIT

```
1      10.0      50.0      2.0      0.0      0.0
Sandia_5m_162.5_RPM_cord_6in_Rmax_2.5m

2      1      .06220      2.04      1.0      6      1      21
      1      1      1      1      1      1      1      1      1      1
      .00194898      3.795e-07      162.5      8.0360
      8.202      33.4      .17      60.0.
```

Input files to MSVAWT

```
Startfile      s5n02.str

SITE1002.WND
CLCDP.DAT
S5N.GEO
162.5
.
```

<i>windfile</i>	site1002.wnd	
21 0.1	200.0	1.005
4.47040	461.0	0.00001817
5.36448	402.0	
6.25856	419.0	
7.15264	385.0	
8.04672	329.0	
8.94080	225.0	
9.83488	212.0	
10.72896	143.0	
11.62304	138.0	
12.51712	96.0	
13.41120	71.0	
14.30528	39.0	
15.19936	21.0	
16.09344	12.0	
16.98752	7.0	
17.88160	3.0	
18.77568	3.0	
19.66976	0.0	
20.56384	2.0	
21.45792	0.0	
22.35200	0.0	
.		

clcdfile *clcdp.dat* (Sample for one Reynolds number)

```

...
NACA0015 LIFT DRAG
11 1000.0 1000000.0 0.15
10000.0 25
0.0 0.0 0.0360
2.0 0.0715 0.0366
4.0 0.0581 0.0383
6.0 -0.0781 0.0400
8.0 -0.1484 0.0644
10.0 -0.0791 0.0910
12.0 0.0138 0.1230
14.0 0.1172 0.1580
16.0 0.2242 0.1960
18.0 0.3319 0.2380
20.0 0.4399 0.2820
22.0 0.5479 0.3290
24.0 0.6559 0.3790
26.0 0.7639 0.4320
30.0 0.8550 0.5700
35.0 0.9800 0.7450
40.0 1.0350 0.9200
50.0 1.0200 1.2150
60.0 0.8750 1.4700
80.0 0.3650 1.7800
100.0 -0.1850 1.7500
120.0 -0.6700 1.4650
140.0 -0.9800 0.9250
160.0 -0.6350 0.3200
180.0 0.0 0.0250
...

```

<i>geomfile</i>	s5n.geo	
1 2.5	20 36	
5.000000	0.152400	0.460402 2
0.193307	0.152400	0.460402 2
0.579921	0.152400	0.460402 2
0.966535	0.152400	0.460402 2
1.353149	0.152400	0.460402 2
1.737180	0.152400	0.433412 2
2.018331	0.152400	0.336091 2
2.204202	0.152400	0.292647 2
2.329166	0.152400	0.269248 2
2.407197	0.152400	0.256445 2
2.444852	0.152400	0.250692 2
2.444852	0.152400	0.250692 2
2.407197	0.152400	0.256445 2
2.329166	0.152400	0.269248 2
2.204202	0.152400	0.292647 2
2.018331	0.152400	0.336091 2
1.737180	0.152400	0.433412 2
1.353149	0.152400	0.460402 2
0.966535	0.152400	0.460402 2
0.579921	0.152400	0.460402 2
0.193307	0.152400	0.460402 2

•

Appendix C Flowind 19 Meter Test Case Input

Input file to SLICEIT

```
10      10.0      60.0      2.0      0.0      0.0
Altamont_19m_52.5_RPM_cord_28in_Rmax_31.5ft

2  1  .07407  2.635  1.0  6  1  21
1  1  1  1  1  1  1  1  1  1  1  1  1
0.001950 3.795e-07  52.5  31.5
15.1      33.0      .17      60.0•
```

Input files to MSVAWT

```
Startfile      a19n21.str

SITE1021.WND
CLCDP.DAT
A19N.GEO
52.5
•
```

```
windfile      site1021.wnd

60  0.1  200.0  1.005  0.00001817
0.44704 383.0
0.89408 427.0
1.34112 381.0
1.78816 295.0
2.23520 281.0
2.68224 297.0
```

3.12928 285.0
3.57632 255.0
4.02336 226.0
4.47040 194.0
4.91744 188.0
5.36448 152.0
5.81152 173.0
6.25856 149.0
6.70560 170.0
7.15264 156.0
7.59968 159.0
8.04672 188.0
8.49376 188.0
8.94080 207.0
9.38784 192.0
9.83488 155.0
10.28192 169.0
10.72896 143.0
11.17600 166.0
11.62304 163.0
12.07008 132.0
12.51712 134.0
12.96416 163.0
13.41120 145.0
13.85824 140.0
14.30528 138.0
14.75232 118.0
15.19936 113.0
15.64640 85.0
16.09344 78.0
16.54048 90.0
16.98752 91.0
17.43456 82.0

17.88160	67.0
18.32864	65.0
18.77568	72.0
19.22272	58.0
19.66976	46.0
20.11680	49.0
20.56384	34.0
21.01088	39.0
21.45792	33.0
21.90496	26.0
22.35200	17.0
22.79904	13.0
23.24608	10.0
23.69312	4.0
24.14016	5.0
24.58720	5.0
25.03424	4.0
25.48128	0.0
25.92832	1.0
26.37536	0.0
26.82240	0.0
.	.

(Not shown - See sample for one Reynolds number Appendix B)

<i>clcdf</i> file	clcdp.dat
-------------------	-----------

geomfile a19n.geo

[illegible]